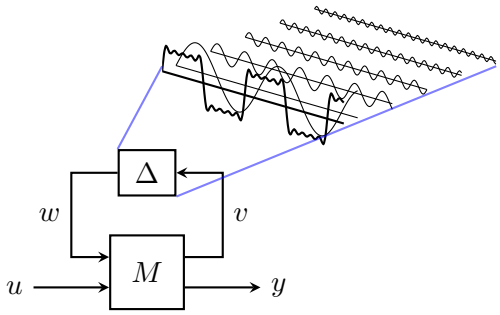
Probabilistic Robustness Analysis with Aerospace Applications



A DISSERTATION SUBMITTED TO
THE FACULTY OF MECHANICAL SCIENCE AND ENGINEERING
OF TECHNISCHE UNIVERSITÄT DRESDEN
BY

Luca Luciano Evangelisti

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE ACADEMIC DEGREE
Doktoringenieur (Dr.-Ing.)

Abstract

This thesis develops theoretical and computational methods for the robustness analysis of uncertain systems. The considered systems are linearized and depend rationally on random parameters with an associated probability distribution. The uncertainty is tackled by applying a polynomial chaos expansion (PCE), a series expansion for random variables similar to the well-known Fourier series for periodic time signals. We consider the linear perturbations around a system's operating point, i.e., reference trajectory, both from a probabilistic and worst-case point of view.

A chief contribution is the polynomial chaos series expansion of uncertain linear systems in linear fractional representation (LFR). This leads to significant computational benefits when analyzing the probabilistic perturbations around a system's reference trajectory. The series expansion of uncertain interconnections in LFR further delivers important theoretical insights. For instance, it is shown that the PCE of rational parameter-dependent linear systems in LFR is equivalent to applying Gaussian quadrature for numerical integration.

We further approximate the worst-case performance of uncertain linear systems with respect to quadratic performance metrics. This is achieved by approximately solving the underlying parametric Riccati differential equation after applying a polynomial chaos series expansion.

The utility of the proposed probabilistic robustness analysis is demonstrated on the example of an industry-sized autolanding system for an Airbus A330 aircraft. Mean and standard deviation of the stochastic perturbations are quantified efficiently by applying a PCE to a linearization of the system along the nominal approach trajectory. Random uncertainty in the aerodynamic coefficients and mass parameters are considered, as well as atmospheric turbulence and static wind shear. The approximate worst-case analysis is compared with Monte Carlo simulations of the complete nonlinear model. The methods proposed throughout the thesis rapidly provide analysis results in good agreement with the Monte Carlo benchmark, at reduced computational cost.

If I have seen further, it is by
standing on the shoulders of giants.

Isaac Newton

Prolog

The present dissertation is the result of three years of research work with my advisor Prof. Harald Pfifer. This thesis would not have been possible without him. I cannot emphasize enough how grateful I am for the exceptional support, high-quality (weekly!) feedback, and for always providing the right guidance to help me navigate through the research outback. Just as Norbert Wiener realized — all intelligent behavior is based on feedback mechanisms — I am profoundly indebted to him for his outstanding mentorship.

Over the last three years, I was affiliated with the Institute of System Dynamics and Control, German Aerospace Center (DLR). I thank Dr. Gertjan Looye for his continuous support and for providing me with a large amount of freedom in my work which was a great enabler for creativity. Thanks also to all my colleagues at DLR who provided a fun and positive working atmosphere. A special thanks goes to the Institute of Robotics and Mechatronics at DLR for letting me use their coffee machine, which provided a significant amount of fuel for this work.

I thank Prof. Manuel Pusch for the advice in the initial phase of my research, helping me survive in the new working environment at the institute and, shortly after, the COVID lockdown in isolation at home. I have to thank Prof. Florian Holzapfel, Institute of Flight System Dynamics, Technische Universität München, for sparking my interest in probabilistic robust control.

On a more personal note, I would like to thank my family. It is difficult to express how lucky I consider myself to have such a loving mother, dedicated father, and twin brother as a true best friend. At the age of 17, after technical boarding school, my grandfather once came from central Italy to Munich as a guest worker in search of a better life. My father has worked very hard to provide me with the best opportunities

in my life. I feel that my education, ultimately symbolized by this academic degree, is a luxurious privilege and one of their greatest gifts.

I dedicate this work to Lisa — to your happiness.

Contents

Notation

δ	parametric uncertainty vector
Δ	uncertain matrix generated by a linear fractional transformation
Δ_Π	projected Δ matrix
$\mathcal{D}$	image space of uncertain parameter vector δ
$\mathcal{F}_l(M, \Delta)$	lower linear fractional transformation w.r.t. nominal matrix M and uncertain matrix Δ
$\mathcal{F}_u(M, \Delta)$	upper linear fractional transformation
G	linear system
H	Hamiltonian matrix
J	quadratic performance function
$\mathcal{J}_L$	Jacobi matrix w.r.t. L-dimensional orthogonal polynomial basis
L_p^2	space of square-integrable random variables
$\mathcal{L}_2[0, T]$	space of square-integrable time signals
M	nominal matrix generated by a linear fractional transformation
$\mathbb{N}_0$	set of natural numbers including zero
$p(\delta)$	probability density function of δ
$R \prec 0$	means that the symmetric matrix R is negative definite
$\mathbb{R}$	field of real numbers
$\mathbb{R}^{n \times m}$	set of n-by-m real matrices
σ	standard deviation
t	time
T	time horizon
u	system input
x	system state

$X \succeq 0$	means that the symmetric matrix X is positive semidefinite
y	system output
ψ_α	orthogonal polynomial basis functions indexed by $\alpha \in \mathbb{N}_0$
Ψ	vector of polynomial basis functions
$\langle \cdot \rangle = \mathbb{E}[\cdot]$	mathematical expectation
$\mathbb{V}[\cdot]$	variance operator
$\otimes$	Kronecker product
cf.	compare (*confer*)
e.g.	for example (*exempli gratia*)
i.e.	that is (*id est*)

It has turned out that it is in principle impossible to know, to measure the position and velocity of a piece of matter with arbitrary accuracy.

Werner Karl Heisenberg

Chapter 1

Introduction

1.1 Background

Uncertainty

This thesis deals with uncertainty in control systems engineering. But what is "uncertainty"? At the heart of every model-based control design lies a mathematical description of the system. The term uncertainty coins the difference between the mathematical model and the real physical system [77, 95]. Uncertainty arises within dynamic systems due to a variety of reasons [74, 84].

- Historically, the presence of uncertainty was already recognized in the work of Bode and Nyquist during the advance of telecommunication technology, requiring the use of feedback control. It was realized that equalizers are necessary to compensate for varying phase and attenuation characteristics of transmission lines, e.g., due to unknown length of the line or varying characteristics attributed to daily temperature changes [17].

- As a modern example from aircraft control, certain parameters are often only known with limited accuracy, such as aerodynamic stability derivatives or air data values. Modeling assumptions may inject further errors, and generally the flow of signals and information to the autopilot may be imprecise [45].

Within this thesis, we consider mathematical approaches to systematically account for uncertainty. Such efforts may be distinguished by the representation or class of

uncertainties considered. Within the classical worst-case paradigm, deterministic set-bounded uncertainty is considered. Another viewpoint is a probabilistic interpretation of the uncertainty. Regardless, throughout various fields of the scientific literature, uncertainty is typically divided into different types. For instance, the mathematical field of uncertainty quantification considers the two classes *aleatoric* and *epistemic* uncertainty, see [77]. The Latin word *alea*, meaning a die, refers to inherently random phenomena, random in the sense of rolling a dice. The Greek meaning for knowledge, $E\pi\iota\sigma\tau\acute{\eta}\mu\eta$, indicates uncertainty attributed to the absence of knowledge. Epistemic uncertainty may be divided further into parametric and non-parametric (model form) uncertainty, as is common within robust control theory.

History of Robust Control

Robust control designates the mathematical effort of explicitly considering uncertainty within controller design. This includes dedicated analysis methods. It should be mentioned that it is not necessarily required to use robust control in order to obtain a robust controller. This was one of the insights of the robust flight control challenge [45]. However, especially in aerospace applications, where uncertainties may be large and the consequences of failure catastrophic, acknowledging the presence of uncertainty is inevitable. To that end, robust control may be viewed as a systematic approach to explicitly account for uncertainty present in the system dynamics.

Reviews of the historical developments and milestones generally distinguish two paradigms: stochastic linear quadratic gaussian (LQG) and $\mathcal{H}_\infty$ worst-case optimal control [64, 56]. The stochastic viewpoint was superseded by the worst-case formalism. Recently, there has been a shift in attention back to a probabilistic point of view [20, 10].

According to [33], around 1950–1960, the United States and the Soviet Union were in a race to space and massively conducting research within guidance, navigation, and control of space vehicles. Driven by the engineering applications of the time, LQG control marked the first major success of optimal control theory. Essential characteristics of the considered systems were linear dynamics disturbed by Gaussian white noise. Since the nature of space vehicles is essentially ballistic, the LQG paradigm turned out to be very effective. Sufficiently accurate mathematical models could be derived. The theoretical achievements by Wiener, Hopf, and Kalman spurred the development of the

well-known Kalman filter within the Apollo program, as well as the establishment of state-space methods.

However, the enthusiasm regarding this era of optimal control was dented in the 1970s as engineers struggled to apply LQG control to further industrial problems. According to [64], LQG controllers surprisingly failed within military aircraft (F-8C Crusader) and submarine (Trident) applications. It was shockingly realized that LQG control lacked robustness guarantees [22]. The optimal control methods of that time were sensitive to uncertainty. Roughly, robustness problems could arise due to the mathematical objective function not properly reflecting the actual optimization problem intertwining uncertainty and feedback control [33, 74, 83]. Rohrs' counterexample showed that adaptive control also suffered from arbitrarily small robustness [61]. The stage was set for the revolutionary starting point of robust control theory.

The driving motivation of the early 1980s was then to make optimal control theory work in practice. In essence, robustness guarantees were developed for all considered uncertainties, including the worst-case. A major milestone was the representation of uncertainty within the $\mathcal{H}_\infty$ paradigm introduced by Zames [94]. Subsequently, the field of robust control flourished. The structured singular value μ was introduced in 1982 [23, 63]. Matlab's Robust Control Toolbox [65, 4] was published in 1988, enabling students and engineers alike to apply robust and optimal control techniques within various fields of engineering. Hence, the road was paved for successful applications within aerospace (helicopter control [58], VSTOL [38] and aeroservoelastic [82] aircraft, missiles [75]), robotics [66], wind turbines [54], and even economics [34]. The theory has generalized to encompass linear parameter-varying [72], finite horizon time-varying [78], periodic [89], and also nonlinear (via integral quadratic constraints [48]) systems.

1.2 Motivation

Probabilistic Robustness Analysis

Following the modern robust control period, the stochastic paradigm has regained interest over the last decades [62, 20, 5, 10]. The reasons for this development are manifold.

Firstly, accounting for the worst-case uncertainty may be overly conservative. Safeguarding against a worst-case that rarely or never occurs is not always the optimal

approach [74, 79]. This brings us to our second point: essentially, classical worst-case analysis neglects an entire "dimension" of the problem. Probabilistic information describing the distribution of the uncertainty within the considered uncertainty set is not considered. Thirdly, specifications of system performance have matured. More often than not, they are given in terms of reliability and associated acceptable risk probabilities. In civil engineering for example, the probable expected lifetime of a building is of interest for owners and insurance companies [47]. Also in the context of automatic landing systems for aircraft, performance requirements are specified in terms of probability levels [9].

As noted earlier, uncertainty may be studied from different points of view and classified by their mathematical representation. It is argued that the probabilistic view is a more natural representation of uncertainty, closer to the real world problems encountered in practice. The idea is to blend model-based predictions with statistical methods, enabling engineers to concentrate on interesting scenarios in a meaningful way, i.e., that are relevant both physically and statistically [77]. This further allows the incorporation of data into the analysis via Bayesian methods.

However, we do not argue with computational reasons here. Note that the introduction of probabilities does not simplify the problem of robustness analysis. On the contrary, it requires accurate evaluation of a potentially high-dimensional probability integral. For this, computational efficiency is a challenge due to the well-known curse of dimensionality. Rather, it is argued that the problem of robustness analysis is hard regardless from which point of view (e.g., the computation of μ is also NP-hard [16]). Therefore, in an industrial context, probabilistic robustness analysis methods can only complement existing methods for Verification and Validation (V&V). There is still value in searching for the worst-case, e.g., via optimization, in order to gain another perspective on the problem.

Polynomial Chaos

It is safe to say that Monte Carlo sampling is the established approach within probabilistic robustness analysis against uncertain system parameters. While generally simple to implement, Monte Carlo estimators are known to converge rather slowly. This means that the standard deviation of the approximation error decreases only asymptotically

proportional to $N^{-1/2}$, N being the sample size. Theoretically, polynomial chaos expansions (PCEs) promise superior — exponential — convergence rates. This holds true especially for smooth parameter-dependencies, a property of Fourier series known as spectral convergence. Being founded on the theory of Fourier, PCEs also offer an intuitive analogy in the sense of generalizing the well-known expansion for periodic time signals of finite energy to random variables of bounded variance.

Recently, following the first applications within systems and control [37, 53], PCEs have received a significant amount of attention within the probabilistic robust control community, e.g., for stochastic linear quadratic regulation [29, 85], linear parameter varying [8], or model predictive [55] control. Generally, the approach is also applicable to nonlinear models and arbitrary non-Gaussian uncertainties [43]. See e.g. [41] for a review of PCE applications within systems and control.

Finite Horizon Robustness

The current state of the art methods for V&V of complex dynamic systems are mostly based on extensive simulation, e.g., within Monte Carlo analysis [91] or nonlinear worst-case optimization [40]. A strength of these methods is their general applicability and ease of use since they treat the system as a black-box. However, this comes at the price of computational efficiency, i.e., the computation time may be very large especially when evaluation of high-fidelity simulation models is costly or in case of rare events. Unfortunately, there is no guarantee of actually finding the global worst-case.

This thesis advances a less time-consuming, complementary approach: analysis methods based on linearization. For uncertain linear time-invariant (LTI) systems, guaranteed stability and performance certificates can be computed efficiently with software such as Matlab's Robust Control Toolbox. These LTI analysis methods are based on linearizing the nonlinear system around a steady-state operating point. However, a large class of engineering applications involves the tracking of a predefined trajectory as a primary incentive. Examples are aircraft [13, 39], space launch vehicles [15], and robotic manipulators [52]. Such systems are more accurately approximated by linearizing the nonlinear dynamics with respect to a reference trajectory defined over a finite time horizon. This yields a finite horizon linear time-varying (LTV) system. Within this thesis, probabilistic robustness analysis methods are developed for such systems.

Regarding the difference between classical LTI approximations and nonlinear simulations, the work can also be motivated by closing the gap between these two analysis paradigms.

1.3 Research Highlights

This thesis develops theoretical and computational methods to analyze the robustness of uncertain linear systems. After introducing the problem formulation underlying the theory of classical robust control (Chapter 2) and polynomial chaos expansions (Chapter 3), the contributions are of both theoretical and practical nature:

1. **Computational Efficient Robustness Analysis of Uncertain LTV Systems in Linear Fractional Representation.** One of the primary research questions addressed within this thesis is *what happens if we apply a polynomial chaos expansion to a linear fractional transformation (LFT)?* See also the picture on the title page. This was the driving idea behind our first work [25]. The first part of Chapter 4 shows that this leads to significant computational benefits when analyzing the stochastic perturbations from a system's reference trajectory due to uncertain model parameters.

2. **Equivalence of LFT Series Expansion and Gaussian Quadrature.** The second part of Chapter 4 is based on an observation made by the author: the eigenvalues of LFT expanded uncertain linear system matrices are realizations of the actual system. This observation can be generalized: it is proven in Section 4.5.2 that the "stochastic modes" of an LFT expanded system are given by certain realizations of the original system — the Gauss quadrature nodes. One of the main theoretical takeaways is that application of a polynomial chaos series expansion to an uncertain linear system with rational parameter-dependence, in linear fractional representation, is equivalent to numerical integration by an approach known as Gaussian quadrature.

3. **Polynomial Chaos Approximation of the Worst-Case Quadratic Performance of Uncertain LTV Systems.** Chapter 5 provides a practical approach to the worst-case analysis of uncertain finite horizon LTV systems with respect to

quadratic performance metrics. The idea is founded on approximately solving the governing parametric Riccati differential equation by applying a polynomial chaos series expansion. The effectiveness of the approach is illustrated by means of an exemplary worst-case performance analysis of a space launcher during atmospheric ascent.

4. **Industrial-Grade Autoland Application Example.** Finally, the probabilistic robustness analysis methods proposed in this thesis are demonstrated on the example of an industry-sized automatic landing system in Chapter 6. The analyzed aircraft model represents an Airbus A330 during final approach and landing. Probabilistic and worst-case perturbations around the nominal reference trajectory are efficiently assessed with respect to uncertainty in the aerodynamic coefficients and mass parameters, atmospheric turbulence, and static wind shear.

1.4 Related Work

In addition to the literature already reviewed throughout this chapter, we briefly point to some existing approaches explicitly related to the contributions of this thesis in the following. The most related work to the polynomial chaos series expansion of a LFT system is [59]. Therein, a transformation is also applied to the linear system before employing a polynomial chaos series expansion. For the transformation proposed in [59], Lyapunov equations need to be solved at the Gauss quadrature points in order to numerically compute the expansion of the transformed linear system. Both [59] and our approach preserve stability of the uncertain linear system. Other related approaches explicitly consider the truncation error of the expansion, see for instance [50, 86]. Convergence guarantees of the expansion applied to Lipschitz differential equations are developed in [2, 3]. Dedicated approaches to overcome an accuracy degradation over time of the series expansion in its original form have also been proposed [32].

There exists a plethora of alternative approaches to approximate the worst-case quadratic performance of uncertain LTV systems. Several algorithms are rooted to bounding the uncertainty within the framework of integral quadratic constraints (IQCs), see for instance [12, 19, 71]. The method in [12] relies on nonlinear optimization whereas [71] iterates between solving a Riccati differential equation and a gridded differential

linear matrix inequality. The aforementioned approaches typically consider the uncertainty as deterministic and yield an upper bound to the true worst-case performance, which may encompass some conservatism.

Robustness analysis along predefined trajectories is required within various engineering applications, such as robotics [52], space launchers [15], missiles [19], or aircraft [39]. The specific autolanding system considered in Chapter 6 is also analyzed in [14] via IQC techniques.

We demand rigidly defined areas of doubt and uncertainty!

The Hitchhiker's Guide to the Galaxy
Douglas Adams

Chapter 2

Classical Robustness

This chapter lays the theoretical foundation for the robustness analysis problem considered throughout this thesis. A key aspect is that the analysis of uncertain linear systems with respect to quadratic performance metrics boils down to solving a Riccati equation. The linear fractional transformation is introduced as an important tool within robust control theory.

2.1 Uncertain Dynamic Systems

2.1.1 Nonlinear Systems

The objective of this thesis is to analyze the effect of parametric uncertainty δ on dynamic systems. Throughout this chapter, we assume the dominant point of view in robust control theory: $\delta \in \mathcal{D} \subset \mathbb{R}^{n_\delta}$ is an unknown parameter vector bounded within a prescribed compact set $\mathcal{D}$ [74]. Consider the following uncertain nonlinear system:

$$\begin{aligned} \dot{x} &= f(x, u, \delta) \\ y &= g(x, u, \delta). \end{aligned} \tag{2.1}$$

It is common in engineering to represent physical systems by such nonlinear state-space models. Therein, the initial condition $x(0) = x_0$ may or may not be uncertain, i.e., additionally depend on δ. $x(t, \delta) \in \mathbb{R}^{n_x}$ is the state, $u(t) \in \mathbb{R}^{n_u}$ the input, and $y(t, \delta) \in \mathbb{R}^{n_y}$ the output. The uncertain parameters δ are assumed to be time-invariant.

Note that Equation (2.1) may be interpreted as a set of systems, parametrized by δ.

Assume the performance of the system (2.1) is measured by some metric. The main challenge is then to check if the specifications hold for all $\delta \in \mathcal{D}$ and find the worst-case performance. For general nonlinear systems (2.1), this is a high-dimensional, non-convex optimization problem. It is a hard problem to solve. Currently, there only exist approximate solution techniques. For example, Monte Carlo simulation [91] or global optimization [40] are often applied. However, for linear systems derived by linearization of Equation (2.1) and specific, e.g., quadratic, performance metrics, the optimization problem is convex, and to put it in the words of [18], "with only a bit of exaggeration, [...] if you formulate a practical problem as a convex optimization problem, then you have solved the original problem". Uncertain but linear systems are the theme of the remainder of this chapter, and a central theme throughout this thesis.

2.1.2 Linear Systems

Linear systems, in contrast to nonlinear systems, are well understood. Consider, for example, one of the most fundamental aspects within the theory of differential equations: existence of a solution. A linear system is always bounded over a compact time interval, whereas nonlinear systems may exhibit the phenomenon of finite escape time. Justifiably, a control systems engineer will not want an implemented controller to grow unbounded over a bounded time interval. Therefore, controllers are often developed as linear systems. The following paragraphs introduce the classes of linear differential equation systems considered in the scope of this thesis. Linear systems theory can be found in any textbook on control, e.g., [33], [74], or [95].

Nominal LTI Systems

Consider a nominal, i.e., certain, linear time-invariant (LTI) system G

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du. \end{aligned} \tag{2.2}$$

Here, A, B, C, and D are (bounded) real matrices, independent of time, and of appropriate dimension. Such linear state-space models are straightforward to obtain from a

nominal nonlinear system [i.e., (2.1) for a specific value of $\delta \in \mathcal{D}$] by linearizing around a stationary operating point[1]. Given an initial condition $x(0) = x_0$ and a specific input $u(t)$, the LTI system response is known to be

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)\,\mathrm{d}\tau. \tag{2.3}$$

Therein, the matrix exponential is $e^{At} := \sum_{k=0}^{\infty} \frac{(At)^k}{k!}$. The output may then be obtained simply by $y(t) = Cx(t) + Du(t)$. The system G is said to be stable, i.e., A is asymptotically stable, if and only if all eigenvalues of A have a strictly negative real part.

In the following, the state-space matrices A, B, C, and D are extended to depend on time t and/or on uncertain parameters δ.

Nominal LTV Systems

Let $A(t)$, $B(t)$, $C(t)$, and $D(t)$ be piecewise-continuous (bounded) functions mapping from $[0, T]$ to a real matrix of appropriate dimension. Such time-varying state-space matrices are readily obtained by linearizing a nonlinear model around a reference trajectory with a finite time horizon $T < \infty$. This yields the definition of a nominal linear time-varying (LTV) system G

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + B(t)u(t) \\ y(t) &= C(t)x(t) + D(t)u(t). \end{aligned} \tag{2.4}$$

Since LTI systems are just a special case of LTV systems, we overload the notation for G. Throughout, the precise meaning will be clear from the context.

Uncertain LTI Systems

Consider the nominal time-invariant case (2.2). Extend the state-space matrices to depend on uncertain parameters δ, e.g., $A : \mathcal{D} \mapsto \mathbb{R}^{n_x \times n_x}$. We obtain an uncertain

[1] Actually, the variables x, u, and y in Equation (2.2) denote different quantities compared to the variables in (2.1). Equation (2.2) represents deviations from the stationary operating point. It is common within control systems engineering to proceed with the same notation for both deviation and absolute variables.

LTI system

$$\begin{aligned} \dot{x} &= A(\delta)x + B(\delta)u \\ y &= C(\delta)x + D(\delta)u. \end{aligned} \tag{2.5}$$

Note that, while nominal LTI systems are readily derived by linearizing a nominal nonlinear system around a single operating point, for uncertain parameter-dependent systems (2.5), the operating point is generally parameter-dependent as well. This difficulty is inherited by uncertain LTV systems introduced subsequently.

Uncertain LTV Systems

In Equation (2.4), additionally to the dependence on time, extend the functions A, B, C, and D by an argument δ. This yields mappings from $[0, T] \times \mathcal{D}$ to a real matrix of appropriate dimension. We obtain an uncertain LTV system

$$\begin{aligned} \dot{x}(t, \delta) &= A(t, \delta)x(t, \delta) + B(t, \delta)u(t) \\ y(t, \delta) &= C(t, \delta)x(t, \delta) + D(t, \delta)u(t). \end{aligned} \tag{2.6}$$

All of the linear systems introduced above can be seen as special cases of the general system (2.6). The following sections are concerned with measuring the worst-case performance of such systems.

Section 2.2 considers nominal performance. Robust performance under parametric uncertainty δ is considered in Section 2.3. Assuming the system is time-varying on a finite horizon or time-invariant on an infinite horizon does not significantly alter the structure of the underlying optimal control problem [33]. Therefore, we start each section with the finite horizon time-varying case. The LTI case then follows as a special case.

2.2 Nominal Performance

There are many ways of characterizing the performance of linear dynamic systems. We concentrate here on quadratic performance indices. Important instances of quadratic performance are, e.g., $\mathcal{H}_\infty$ constraints, strict passivity, and sector bounds [67]. It is

well known that the worst-case performance of linear systems with quadratic cost functionals can be posed as an optimal control problem involving the solution of a Riccati equation [33]. Riccati equations are at the core of countless quintessential problems within control systems engineering, e.g., yielding necessary analysis and synthesis conditions [95], within applied mathematics as solutions to variational problems [1], and beyond [34]. Note that also other equivalent performance analysis frameworks exist, for example based on linear matrix inequalities (LMIs) which can be quite general and flexible. However, LMIs are out of the scope of this thesis, and the reader is referred to the literature for details ([68] is recommended).

Finite Horizon Case

Denote by $\mathcal{L}_2[0,T]$ the Hilbert space of all signals, e.g., $f(t)$, with bounded norm $\|f\|_{\mathcal{L}_2[0,T]} = \left\{ \int_0^T f^T(t) f(t)\,\mathrm{d}t \right\}^{1/2} < \infty$. Consider the quadratic cost $J : \mathcal{L}_2[0,T] \mapsto \mathbb{R}$

$$J(u) = x^T(T)Q(T)x(T) + \int_0^T \begin{bmatrix} x(t) \\ u(t) \end{bmatrix}^T \begin{bmatrix} Q(t) & S(t) \\ S^T(t) & R(t) \end{bmatrix} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \mathrm{d}t. \tag{2.7}$$

The performance J is parametrized by piecewise-continuous (bounded) matrix functions $Q = Q^T : [0,T] \mapsto \mathbb{R}^{n_x \times n_x}$, $S : [0,T] \mapsto \mathbb{R}^{n_x \times n_u}$, and $R = R^T : [0,T] \mapsto \mathbb{R}^{n_u \times n_u}$. If we interpret $u(t)$ as an external disturbance and $y(t)$ as a generic error signal, note that worst-case quadratic performance analysis of nominal LTV systems with $x(0) = 0$ can be formulated as the following optimization problem [71]:

$$\begin{aligned} \max_{u \in \mathcal{L}_2[0,T]} \quad & J(u) \\ \text{subject to} \quad & \text{Equation (2.4)},\ x(0) = 0,\ \text{and } \|u\|_{\mathcal{L}_2[0,T]} = 1. \end{aligned} \tag{2.8}$$

As a specific instance of (2.8), consider the finite-horizon $\mathcal{L}_2$-to-Euclidean gain of G defined in [71] by

$$\|G\|_{2,\mathcal{L}_2[0,T]} = \sup \left\{ \frac{\|y(T)\|_2}{\|u\|_{\mathcal{L}_2[0,T]}} \;\middle|\; u \in \mathcal{L}_2[0,T] \setminus 0,\ x(0) = 0 \right\}. \tag{2.9}$$

Assuming $D(T) = 0$ for well-posedness, the finite-horizon $\mathcal{L}_2$-to-Euclidean gain is bounded by $\|G\|_{2,\mathcal{L}_2[0,T]} \leq \gamma$, for some $\gamma > 0$, if and only if

$$J(u) \leq 0 \qquad \forall u \in \mathcal{L}_2[0,T]. \tag{2.10}$$

Therein, $J(u)$ is parametrized by $Q(t) = 0\ \forall t \in [0,T)$, $Q(T) = C^T(T)C(T)$, $S(t) = 0$, and $R(t) = -\gamma^2 I_{n_u}$. Other system gains can be defined similarly to (2.9), depending on the chosen system norms, such as the finite-horizon induced $\mathcal{L}_2$-gain[2]. See [33, 71] for further information and [69] for a generalization to non-zero initial conditions.

The optimal control problem (2.8) is inherently related to the considered worst-case gain due to Equation (2.10). The following theorem states an equivalence between bounded quadratic performance and existence of the solution to a Riccati differential equation (RDE).

Theorem 2.1 (LTV Bounded Real Lemma [71])**.** *Let the quadratic cost* (2.7) *be specified by given $Q(t)$, $S(t)$, and $R(t)$; with $R(t) \prec 0\ \forall t \in [0,T]$. Then the following statements are equivalent:*

1. *$\exists \epsilon > 0$ such that $J(u) \leq -\epsilon \|u\|^2_{\mathcal{L}_2[0,T]}$ for all $u \in \mathcal{L}_2[0,T]$ and $x(0) = 0$.*
2. *The solution $X(t)$ of the Riccati differential equation*

$$\begin{aligned} -\dot{X} &= A^T(t)X + XA(t) + Q(t) - \big[XB(t) + S(t)\big]R^{-1}(t)\big[XB(t) + S(t)\big]^T \\ X(T) &= Q(T) \end{aligned} \tag{2.11}$$

   exists on $[0,T]$.

This theorem characterizes quadratic performance of a nominal LTV system (2.4) in a manner that gives a necessary condition for the quadratic performance function J to be strictly negative for all square-integrable trajectories of the system. Note that for $T \to \infty$ and time-invariant coefficients, if $X(t)$ exists it converges towards the steady-state solution of the RDE, which is the stabilizing solution of the associated algebraic Riccati equation, see [1] or [33]. The time-invariant infinite horizon case is addressed in the following section.

[2]For stable LTI systems on an infinite horizon, the induced $\mathcal{L}_2$-gain is equivalent to the $\mathcal{H}_\infty$-norm.

Infinite Horizon Case

In this section, we briefly state quadratic performance analysis conditions for nominal LTI systems over an infinite time horizon. The presented results are well-known within the robust control literature and can be found in, e.g., [33] or [95].

Abbreviate $\mathcal{L}_2[0,\infty)$ simply by $\mathcal{L}_2$. Consider the quadratic cost $J : \mathcal{L}_2 \mapsto \mathbb{R}$ parametrized by time-invariant matrices $Q = Q^T \in \mathbb{R}^{n_x \times n_x}$, $S \in \mathbb{R}^{n_x \times n_u}$, and $R = R^T \in \mathbb{R}^{n_u \times n_u}$:

$$J(u) = \int_0^\infty \begin{bmatrix} x(t) \\ u(t) \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \mathrm{d}t. \tag{2.12}$$

The following theorem gives analysis conditions for bounded quadratic performance of nominal LTI systems (2.2) over an infinite time horizon with $x(0) = 0$.

Theorem 2.2 (LTI Bounded Real Lemma)**.** *Let the quadratic cost* (2.12) *be given, including Q, S, and $R \prec 0$. Assume that A is asymptotically stable. The following statements are equivalent:*

1. *$\exists \epsilon > 0$ such that $J(u) \leq -\epsilon \|u\|_{\mathcal{L}_2}^2$ for all $u \in \mathcal{L}_2$ and $x(0) = 0$.*

2. *There exists a positive semi-definite solution $X \succeq 0$ to the algebraic Riccati equation*

$$0 = A^T X + XA + Q - (XB + S)R^{-1}(XB + S)^T \tag{2.13}$$

   such that $A - BR^{-1}(B^T X + S^T)$ is asymptotically stable.

3. *The Hamiltonian matrix*

$$H = \begin{bmatrix} A - BR^{-1}S^T & -BR^{-1}B^T \\ -(Q - SR^{-1}S^T) & -(A - BR^{-1}S^T)^T \end{bmatrix} \tag{2.14}$$

   has no eigenvalue on the imaginary axis.

Associated with every algebraic Riccati equation (ARE) is a Hamiltonian matrix, see e.g. [95]. The Hamiltonian is a system matrix for the Hamiltonian dynamics associated with the underlying linear-quadratic optimal control problem [cf. Equation (2.8)]. Note that checking if the stabilizing solution to the ARE (2.13) exists constitutes merely one view of the LTI quadratic performance analysis problem. The analysis condition

involving the Hamiltonian is displayed in Theorem 2.2 for later reference. In practice, the spectrum of the Hamiltonian is often analyzed and at the core of many algorithms [95].

A prominent example of quadratic performance characterized by Theorem 2.2 is the $\mathcal{H}_\infty$-norm. Set $Q = C^T C$, $S = C^T D$, and $R = D^T D - \gamma^2 I \prec 0$ for some $\gamma > 0$. The $\mathcal{H}_\infty$ performance of nominal LTI systems (2.2) can then be bounded by $\|G\|_\infty < \gamma$ if and only if a corresponding item of Theorem 2.2 holds.

2.3 Robust Performance

It is straightforward to generalize the nominal performance analysis conditions introduced in Section 2.2 to robust performance under parametric uncertainty $\delta \in \mathcal{D}$. All involved functions are simply extended by an additional argument δ, similar to the step from nominal LTV systems to uncertain LTV systems in Section 2.1.2. The quadratic performance analysis conditions are generalized analogously: in essence, existence of a parameter-dependent solution to the considered Riccati equation is required for all $\delta \in \mathcal{D}$.

Finite Horizon Case

First, consider the finite horizon time-varying case. The quadratic cost (2.7) is generalized to yield the generic mapping $J : \mathcal{L}_2[0,T] \times \mathcal{D} \mapsto \mathbb{R}$

$$J(u,\delta) = x^T(T,\delta)Q(T,\delta)x(T,\delta) + \int_0^T \begin{bmatrix} x(t,\delta) \\ u(t) \end{bmatrix}^T \begin{bmatrix} Q(t,\delta) & S(t,\delta) \\ S^T(t,\delta) & R(t,\delta) \end{bmatrix} \begin{bmatrix} x(t,\delta) \\ u(t) \end{bmatrix} \mathrm{d}t. \quad (2.15)$$

Herein, Q, S, and R are extended by an additional argument δ to map from $[0,T] \times \mathcal{D}$ to a real matrix of appropriate dimension, e.g., $Q = Q^T : [0,T] \times \mathcal{D} \mapsto \mathbb{R}^{n_x \times n_x}$. In the interest of better readability, define the RDE coefficient matrix $E(t,\delta)$ by

$$E := \begin{bmatrix} Q - SR^{-1}S^T & A^T - SR^{-1}B^T \\ A - BR^{-1}S^T & -BR^{-1}B^T \end{bmatrix}. \quad (2.16)$$

Therein, all involved matrix functions have the same arguments in general. This allows to factorize RDEs of the form (2.11) as in the following robust version of Theorem 2.1.

Theorem 2.3 (LTV Robust Bounded Real Lemma)**.** *Let the generic quadratic cost* (2.15) *be specified by* $Q(t,\delta)$*,* $S(t,\delta)$*, and* $R(t,\delta)$*; with* $R(t,\delta) \prec 0$ *for all* $(t,\delta) \in [0,T] \times \mathcal{D}$*. Then the following statements are equivalent:*

1. $\exists \epsilon > 0$ *such that* $J(u,\delta) \leq -\epsilon \|u\|^2_{\mathcal{L}_2[0,T]}$ *for all* $u \in \mathcal{L}_2[0,T]$*,* $\delta \in \mathcal{D}$*, and* $x(0) = 0$*.*

2. *The solution* $X(t,\delta)$ *of the parameter-dependent Riccati differential equation*

$$\begin{aligned} -\dot{X}(t,\delta) &= \begin{bmatrix} I_{n_x} \\ X(t,\delta) \end{bmatrix}^T E(t,\delta) \begin{bmatrix} I_{n_x} \\ X(t,\delta) \end{bmatrix} \\ X(T,\delta) &= Q(T,\delta) \end{aligned} \tag{2.17}$$

exists for all $(t,\delta) \in [0,T] \times \mathcal{D}$*.*

We omit the proof since it merely extends the proof of Theorem 2.1 conducted in [71] to hold for all $\delta \in \mathcal{D}$. In short, Theorem 2.3 characterizes the quadratic performance of uncertain LTV systems (2.6). It is highlighted that *nominal* quadratic performance according to Theorem 2.1 is straightforward to check by numerical integration of the RDE (2.11). However, certifying *robust* quadratic performance is computationally more challenging, since existence of the solution $X(t,\delta)$ needs to be checked for all values of δ. The situation is similar in the infinite horizon case.

Infinite Horizon Case

For completeness, we briefly present the infinite horizon time-invariant case below. Similar to the previous section, the quadratic cost (2.12) is extended to the generic mapping $J : \mathcal{L}_2 \times \mathcal{D} \mapsto \mathbb{R}$ with time-invariant matrices $Q(\delta)$, $S(\delta)$, and $R(\delta)$. The parameter-dependent Hamiltonian matrix $H(\delta)$ is given by

$$H = \left[\begin{array}{c|c} H_{11} & H_{12} \\ \hline H_{21} & H_{22} \end{array}\right] = \left[\begin{array}{c|c} A - BR^{-1}S^T & -BR^{-1}B^T \\ \hline -(Q - SR^{-1}S^T) & -(A - BR^{-1}S^T)^T \end{array}\right] \tag{2.18}$$

where the dependency on δ is omitted for brevity. This allows to write AREs of form (2.13) in their factorized version, as within the following robust analog of Theorem 2.2.

Theorem 2.4 (LTI Robust Bounded Real Lemma)**.** *Let $A(\delta)$ be asymptotically stable and the generic quadratic cost $J(u,\delta)$ be given, including $Q(\delta)$, $S(\delta)$, and $R(\delta) \prec 0$, for all $\delta \in \mathcal{D}$. The following statements are equivalent:*

1. *$\exists \epsilon > 0$ such that $J(u,\delta) \leq -\epsilon \|u\|_{\mathcal{L}_2}^2$ for all $u \in \mathcal{L}_2$, $\delta \in \mathcal{D}$, and $x(0) = 0$.*

2. *For all $\delta \in \mathcal{D}$, there exists an $X(\delta) \succeq 0$ satisfying the parameter-dependent algebraic Riccati equation*

$$0 = \begin{bmatrix} X(\delta) \\ -I_{n_x} \end{bmatrix}^T H(\delta) \begin{bmatrix} I_{n_x} \\ X(\delta) \end{bmatrix} \tag{2.19}$$

   such that $H_{11}(\delta) + H_{12}(\delta)X(\delta)$ is asymptotically stable.

3. *The Hamiltonian matrix* (2.18) *has no eigenvalue on the imaginary axis for all $\delta \in \mathcal{D}$.*

As noted in the previous section, certifying robust quadratic performance is a hard problem. Existence of the solution to a parameter-dependent Riccati equation needs to be checked for all $\delta \in \mathcal{D}$. Therefore, practical approaches focus on approximations, i.e., lower and upper bounds of the worst-case robust performance.

A large body of results within robust control exists after "pulling out the uncertainty" by applying a linear fractional transformation:

- In the infinite horizon case, if performance is measured by the $\mathcal{H}_\infty$-norm, this then boils down to a structured singular value computation, see e.g. [95].

- In the finite horizon case, analysis conditions involving the solution of an extended RDE can be derived by bounding the uncertainty with integral quadratic constraints, see e.g. [71] or [11].

The linear fractional transformation will also prove to be instrumental within Chapter 4 of this thesis. It is defined in the following.

2.4 Linear Fractional Transformation

This section introduces an important matrix function: the linear fractional transformation (LFT). Several relevant control problems can be expressed in the language of

LFTs and a variety of optimization problems are solved using the LFT framework, see for example [33].

To set the scene, consider for instance the uncertain LTV system (2.6). If the state-space matrices A, B, C, and D depend rationally on δ, it is known that the general system (2.6) can always be written in linear fractional representation[3] (LFR)

$$\begin{aligned} \begin{bmatrix} \dot{x}(t,\delta) \\ v(t,\delta) \\ y(t,\delta) \end{bmatrix} &= \begin{bmatrix} A_M(t) & B_w(t) & B_u(t) \\ C_v(t) & D_{vw}(t) & D_{vu}(t) \\ C_y(t) & D_{yw}(t) & D_{yu}(t) \end{bmatrix} \begin{bmatrix} x(t,\delta) \\ w(t,\delta) \\ u(t) \end{bmatrix} \\ w(t,\delta) &= \Delta(\delta) v(t,\delta) \end{aligned} \tag{2.20}$$

where Δ depends *linearly* on δ. The underlying transformation is defined more formally below. The presented material is based mainly on Chapter 10 in [95], to which the reader is referred for a more detailed exposition.

Definition 2.1. *Let M be a partitioned matrix*

$$M = \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right] \in \mathbb{R}^{(n_\Delta + n_y)\times(n_\Delta + n_u)}. \tag{2.21}$$

Then the upper LFT with respect to $\Delta \in \mathbb{R}^{n_\Delta \times n_\Delta}$ is defined as the matrix function $\mathcal{F}_u(M,\cdot) : \mathbb{R}^{n_\Delta \times n_\Delta} \mapsto \mathbb{R}^{n_y \times n_u}$

$$\mathcal{F}_u(M,\Delta) := M_{22} + M_{21}\Delta(I - M_{11}\Delta)^{-1}M_{12}, \tag{2.22}$$

provided $(I - M_{11}\Delta)$ is invertible.

For parametric uncertainty, as considered throughout this thesis, note that Δ is always a diagonal matrix depending affinely on δ

$$\Delta(\delta) = \begin{bmatrix} \delta_1 I_{m_1} & & 0 \\ & \ddots & \\ 0 & & \delta_{n_\delta} I_{m_{n_\delta}} \end{bmatrix}. \tag{2.23}$$

Therein, each δ_i is possibly repeated with multiplicity m_i, where $n_\Delta = \sum_{i=1}^{n_\delta} m_i$ denotes

[3] It is common to abbreviate the LFT representation of an uncertain model by the short term LFR.

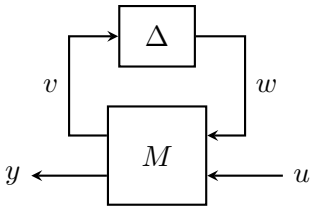


Figure 2.1: Upper LFT

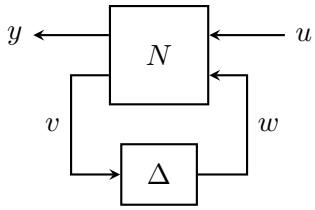


Figure 2.2: Lower LFT

the order of the LFT. The upper LFT is depicted in Figure 2.1. Evidently, it represents the set of equations

$$\begin{bmatrix} v \\ y \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}, \qquad w = \Delta v. \tag{2.24}$$

It is straightforward to verify that $\mathcal{F}_u(M, \Delta)$ is the transformation (2.22) obtained after closing the upper Δ-loop in Figure 2.1, i.e., $y = \mathcal{F}_u(M, \Delta)u$. The lower LFT is defined in a similar fashion according to Figure 2.2.

Definition 2.2. *Denote by*

$$N = \left[\begin{array}{c|c} N_{11} & N_{12} \\ \hline N_{21} & N_{22} \end{array}\right] \in \mathbb{R}^{(n_y+n_\Delta)\times(n_u+n_\Delta)} \tag{2.25}$$

a suitably partitioned matrix. The lower LFT with respect to Δ is the mapping

$$\mathcal{F}_l(N, \Delta) := N_{11} + N_{12}\Delta(I - N_{22}\Delta)^{-1}N_{21}, \tag{2.26}$$

provided the inverse $(I - N_{22}\Delta)^{-1}$ exists.

The equations governing the lower LFT are

$$\begin{bmatrix} y \\ v \end{bmatrix} = \begin{bmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{bmatrix} \begin{bmatrix} u \\ w \end{bmatrix}, \qquad w = \Delta v. \tag{2.27}$$

It is again easy to verify that closing the lower Δ-loop, see Figure 2.2, yields $y = \mathcal{F}_l(N, \Delta)u$.

It is insightful to interpret an LFT, e.g., $\mathcal{F}_l(N, \Delta)$, as a nominal part N_{11} (obtained by setting $\Delta = 0$) and a perturbation term. The matrices N_{12}, N_{21}, and N_{22} shape the perturbation, i.e., how Δ affects the nominal mapping N_{11} [95]. Intuitively, if N is considered as a proper transfer function matrix, a LFT represents the closed-loop transfer matrix from u to y, see Figure 2.2. Therein, N symbolizes the plant and Δ the model uncertainty or controller.

Finally, note that constructing LFTs of minimal order is known to be a problem, see for instance [35] or [57]. However, practical toolboxes are available, such as [36] or Matlab's Robust Control Toolbox.

The mark of a mature, psychologically healthy mind is indeed the ability to live with uncertainty and ambiguity, but only as much as there really is.

JULIAN BAGGINI

Chapter 3

Polynomial Chaos Theory

3.1 Uncertainty Quantification

Consider again the uncertain nonlinear system (2.1) with parameter vector δ introduced in the beginning of Chapter 2. In the mathematical field of uncertainty quantification, the parameters are assumed to be distributed with respect to some known probability density function (pdf). This may result, for instance, from the application of a system identification method or from the incorporation of observed data via Bayesian methods. Thus, the entries of the parameter vector δ_i in Chapter 2 are now replaced by independent random variables for $i = 1, ..., n_\delta$. This renders δ a real random vector. Uncertainty quantification then views the system's response surface as a stochastic process: a mapping $y : [0, \infty) \times \mathcal{D}$ such that for each point in time $y(t, \cdot)$ is a random vector. This is illustrated in Figure 3.1, for the case of a scalar-valued δ and a scalar output y.

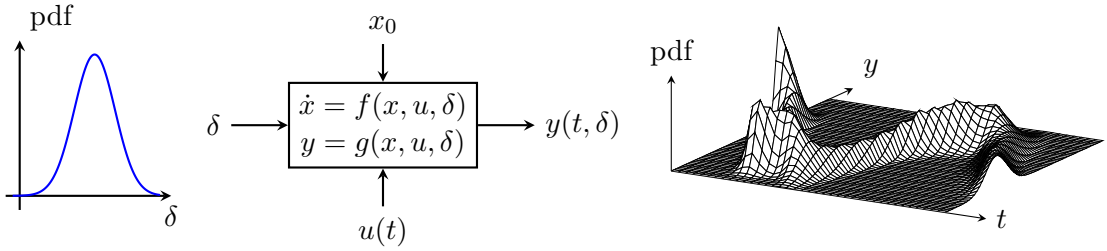


Figure 3.1: Uncertainty Quantification of a Stochastic Process

Roughly, the chief question is as follows. Given an input probability distribution for the random vector δ, what is the distribution of the output y over time? This problem is referred to as propagating the uncertainty (i.e., uncertainty propagation) see [77].

Conceptually one of the simplest approximate solutions to this problem is given by the method of Monte Carlo (MC) [43]. Therein, the probability law and statistics of $y(t,\cdot)$ are inferred from random samples of δ. For example, the mathematical expectation $\mathbb{E}[y(t,\cdot)]$ can be estimated with N independent identically distributed (i.i.d.) samples $\delta^{(i)}$ by

$$\mathbb{E}[y(t,\cdot)] \approx \frac{1}{N}\sum_{i=1}^{N} y(t,\delta^{(i)}). \tag{3.1}$$

Simplicity is both strength and weakness of the MC approach. On the one hand, the system (2.1) is treated as a black-box and solely needs to be evaluated for N realizations of δ. Thus, the original solver of the deterministic problem, for instance, a numerical integrator based on a Runge-Kutta scheme, can be used as is and does not have to be altered. On the other hand, the convergence rate of MC estimators can be rather slow. Specifically, the estimator (3.1) has an estimation error with standard deviation decreasing asymptotically proportional to $N^{-1/2}$. Several sampling strategies have been proposed to partly increase the convergence rate, see [43] or [77].

Within this chapter, we introduce an approach promising superior, exponential, convergence of the mean-square error: the polynomial chaos series expansion. This is a generalized Fourier series for random variables of finite variance. It is mathematically analogous to the Fourier series for periodic time signals of finite energy. The exponential convergence rate of the mean-square approximation error is a central characteristic for expansions of this type. This is known as spectral convergence. Instead of projecting on a harmonic basis, the polynomial chaos expansion approximates functions within a suitable orthogonal polynomial basis of the random parameters. As such, "the canvas" [49] of polynomial chaos is woven substantially out of orthogonal polynomial theory. Therefore, the following section concentrates on building a fundamental understanding of orthogonal polynomials. Subsequently, Section 3.3 formally introduces the polynomial chaos series expansion. The remainder is dedicated to actually computing the expansion coefficients in practice. Two classes of methods are distinguished: intrusive Galerkin projection (Section 3.4) and non-intrusive approaches (Section 3.5).

3.2 Orthogonal Polynomials

This section gives an introduction to orthogonal polynomials. The presented material is based mainly on Chapter 8 in [77]. See also [31] for further information.

3.2.1 Preliminaries

Orthogonal polynomials are fundamentally anchored within practical algorithms for numerical integration based on quadrature rules, see Section 3.5. They also form an important building block of function approximation theory. The most important use-case of orthogonal polynomials in the scope of this thesis is that they provide an orthogonal basis for the Hilbert function space consisting of all random variables with finite variance. This is stated more clearly in the following.

In the interest of clarity, throughout Section 3.2, it is assumed $n_\delta = 1$, i.e., $\delta \in \mathcal{D} \subset \mathbb{R}$ is a scalar. We generalize to the multivariate case $n_\delta > 1$ in Section 3.3.2. Here, we consider δ as a $\mathbb{R}$-valued random variable with probability density $p(\delta)$. Recall that a random variable is nothing else than a measurable function with an associated probability density function [43, 77]. Note that a generic mapping $f(\delta)$ is also a random variable whose probability density can be written in terms of p and f. We consider throughout the Hilbert space of all square-integrable random variables f (i.e., mappings)

$$L_p^2 := \Big\{ f : \mathcal{D} \mapsto \mathbb{R} \;\Big|\; f \text{ measurable and } \|f\|_{L_p^2} < \infty \Big\}. \tag{3.2}$$

Therein, for any $f, g \in L_p^2$ the inner product is defined by

$$\langle f, g \rangle_{L_p^2} = \int_{\mathcal{D}} f(\delta) g(\delta) p(\delta) \,\mathrm{d}\delta = \mathbb{E}[fg]. \tag{3.3}$$

If $\langle f, g \rangle_{L_p^2} = 0$ it is said that f is orthogonal to g. For $f = g$, the induced norm of f is given by

$$\|f\|_{L_p^2} = \sqrt{\langle f, f \rangle_{L_p^2}}. \tag{3.4}$$

We henceforth write $\langle \cdot, \cdot \rangle_{L_p^2} = \langle \cdot, \cdot \rangle$ for notational convenience. In order to highlight the connection with orthogonality, the expectation is abbreviated by $\langle \cdot \rangle = \mathbb{E}[\cdot]$ when appropriate.

For $\alpha \in \mathbb{N}_0$, let $\psi_\alpha : \mathcal{D} \mapsto \mathbb{R}$ be a real polynomial

$$\psi_\alpha(\delta) = 1 + c_1\delta + \ldots + c_\alpha\delta^\alpha \tag{3.5}$$

with scalar coefficients $c_1, ..., c_\alpha \in \mathbb{R}$ (it is assumed $c_0 = 1$). The polynomial degree of ψ_α is α. Now the stage is set for introducing orthogonal decompositions of L_p^2.

Definition 3.1. *Let $p(\delta)$ be a probability density function on $\mathcal{D}$. The set*

$$\{\psi_\alpha \mid \alpha \in \mathbb{N}_0\} \subset L_p^2 \tag{3.6}$$

is referred to as the orthogonal polynomials with respect to $p(\delta)$ if

$$\langle \psi_\alpha, \psi_\beta \rangle = 0 \iff \alpha \neq \beta \quad \forall \alpha, \beta \in \mathbb{N}_0. \tag{3.7}$$

If in addition, $\|\psi_\alpha\|_{L_p^2} = 1$ for all $\alpha \in \mathbb{N}_0$, the ψ_α are called the orthonormal *polynomials*

$$\langle \psi_\alpha, \psi_\beta \rangle = \delta_{\alpha\beta} := \begin{cases} 0 & \text{if } \alpha \neq \beta \\ 1 & \text{if } \alpha = \beta \end{cases} \qquad \forall \alpha, \beta \in \mathbb{N}_0 \tag{3.8}$$

where $\delta_{\alpha\beta}$ denotes the Kronecker delta.

We will mostly consider sets of orthogonal polynomials that form a basis[1] of L_p^2. Examples of such orthogonal polynomials are given below, taken from [77]. Throughout this thesis, Greek letters $(\alpha, \beta, ...)$ are used to index within an orthogonal polynomial basis.

Example 3.1. *The Hermite polynomials* He_α *are the orthogonal polynomials with respect to the standard Gaussian distribution $p(\delta) = (2\pi)^{-1/2} e^{-\delta^2/2}$ on $\mathbb{R}$:*

$$\langle \mathrm{He}_\alpha, \mathrm{He}_\beta \rangle = \int_{-\infty}^{\infty} \mathrm{He}_\alpha(\delta)\mathrm{He}_\beta(\delta) \frac{1}{\sqrt{2\pi}} e^{-\delta^2/2} \, \mathrm{d}\delta = \alpha! \delta_{\alpha\beta}. \tag{3.9}$$

The first few Hermite polynomials are displayed in Figure 3.2.

[1] A set of orthogonal polynomials forms a complete orthogonal basis of L_p^2 under mild additional assumptions [24], e.g., that $\mathcal{D}$ is a compact set.

Example 3.2. *The Legendre polynomials* Le_α *are the orthogonal polynomials with respect to the uniform distribution on* $[-1, 1]$*:*

$$\langle \mathrm{Le}_\alpha, \mathrm{Le}_\beta \rangle = \int_{-1}^{1} \mathrm{Le}_\alpha(\delta)\mathrm{Le}_\beta(\delta)\,\mathrm{d}\delta = \frac{2}{2\alpha+1}\delta_{\alpha\beta}. \tag{3.10}$$

The Legendre polynomials are displayed in Figure 3.3.

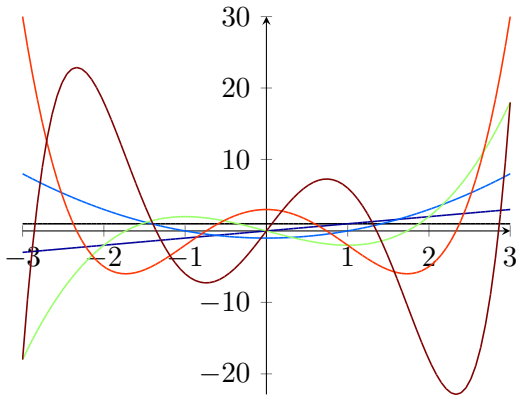


Figure 3.2: Hermite Polynomials

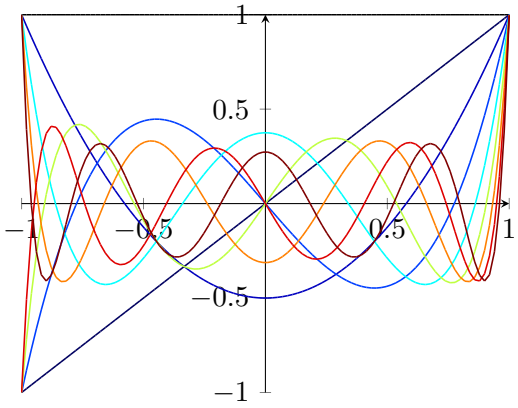


Figure 3.3: Legendre Polynomials

The respective orthogonal polynomials can be constructed for arbitrary continuous distributions of δ, see [24]. In practice, a numerically stable way of constructing them is, for instance, via their three-term recurrence relations. This vital property is introduced in the subsequent section.

3.2.2 Recurrence Relations

An important property of orthogonal polynomials also within numerical algorithms is that they can be constructed recursively[2]:

$$\begin{aligned} \psi_{\alpha+1}(\delta) &= (\delta - a_\alpha)\psi_\alpha(\delta) - b_\alpha\psi_{\alpha-1}(\delta) \\ \psi_0(\delta) &= 1 \\ \psi_{-1}(\delta) &= 0 \end{aligned} \tag{3.11}$$

[2]Strictly speaking, the recursion (3.11) holds for the monic orthogonal polynomials ψ_α. Monic polynomials are polynomials (3.5) with $c_\alpha = 1$. The generalization to any system of orthogonal polynomials is minor and not relevant for the purposes of this thesis.

where a_α and b_α are real scalar coefficients, see [77]. For example, the Hermite and Legendre polynomials are given by the recurrence relations

$$\begin{aligned} \mathrm{He}_{\alpha+1}(\delta) &= \delta \mathrm{He}_\alpha(\delta) - \alpha \mathrm{He}_{\alpha-1}(\delta), \\ \mathrm{Le}_{\alpha+1}(\delta) &= \frac{2\alpha+1}{\alpha+1}\delta \mathrm{Le}_\alpha(\delta) - \frac{\alpha}{\alpha+1}\mathrm{Le}_{\alpha-1}(\delta). \end{aligned} \tag{3.12}$$

Given the recursion coefficients of orthogonal polynomials, their roots, i.e., δ such that $\psi_\alpha(\delta) = 0$, can be readily computed. This is particularly important for numerical integration by Gaussian quadrature, see Section 3.5.

3.2.3 Roots

In this section, it is shown that the roots of orthogonal polynomials are the distinct eigenvalues of a certain matrix. This will play an important role within the proof of our main theoretical results in Chapter 4, Section 4.5.

Consider the vector

$$\Psi(\delta) = [\psi_0(\delta), \psi_1(\delta), \ldots, \psi_{L-1}(\delta)]^T \in \mathbb{R}^L \tag{3.13}$$

concatenating vertically the polynomial basis functions ψ_α up until $\alpha = L-1$. Then the first L recurrence relations (3.11) can be stated in matrix-vector form

$$\delta\Psi(\delta) = \mathcal{J}_L\Psi(\delta) + \psi_L(\delta)[0, ..., 0, 1]^T \tag{3.14}$$

where $\mathcal{J}_L$ is the tridiagonal *Jacobi matrix*

$$\mathcal{J}_L := \begin{bmatrix} a_0 & 1 & & & \\ b_1 & a_1 & 1 & & \\ & b_2 & \ddots & \ddots & \\ & & \ddots & a_{L-2} & 1 \\ & & & b_{L-1} & a_{L-1} \end{bmatrix}. \tag{3.15}$$

Equation (3.14) implies that the roots of the orthogonal polynomial $\psi_L(\delta)$ are the eigenvalues of $\mathcal{J}_L$ with corresponding eigenvectors $\Psi(\delta)$. It is known further that the eigenvalues of $\mathcal{J}_L$ are all real and distinct, i.e., the roots of $\psi_L(\delta)$ are simple, see the following result.

Lemma 3.1. *For every $\alpha \in \mathbb{N}_0$, the orthogonal polynomial $\psi_\alpha(\delta)$ has precisely α distinct real roots $\delta^{(1)}, ..., \delta^{(\alpha)} \in \mathcal{D}$.*

Proof. The proof can be found, e.g., in [31] or [77]. For $\alpha = 0$, by convention $\psi_0 = 1$, see Equation (3.5). For $\alpha \geq 1$, note that

$$\langle \psi_\alpha \rangle = \mathbb{E}[\psi_\alpha] = \int_{\mathcal{D}} \psi_\alpha(\delta) p(\delta) \, \mathrm{d}\delta = 0 \tag{3.16}$$

since $\langle \psi_\alpha, 1 \rangle = \langle \psi_\alpha, \psi_0 \rangle = 0$ due to orthogonality of the basis (3.7). Therefore, ψ_α changes sign at least once in $\mathcal{D}$ and has at least one real root $\delta^{(1)} \in \mathcal{D}$. For $\alpha > 1$, we show in the following by contradiction that there must exist another distinct root $\delta^{(2)} \in \mathcal{D}$ of ψ_α. Assume ψ_α has only $k = 1 < \alpha$ root. Since $\{\psi_\beta \mid \beta \in \mathbb{N}_0\}$ forms a basis for L_p^2, the degree k polynomial $\prod_{i=1}^k (\delta - \delta^{(i)})$ can be represented exactly by the basis polynomials ψ_β, $\beta = 0, ..., k$:

$$\prod_{i=1}^{k} (\delta - \delta^{(i)}) = \sum_{\beta=0}^{k} c_\beta \psi_\beta(\delta) \tag{3.17}$$

with scalar coefficients $c_\beta \in \mathbb{R}$. Therefore, due to orthogonality

$$\Big\langle \psi_\alpha, \prod_{i=1}^{k} (\delta - \delta^{(i)}) \Big\rangle = \langle \psi_\alpha, \sum_{\beta=0}^{k} c_\beta \psi_\beta \rangle = \sum_{\beta=0}^{k} c_\beta \langle \psi_\alpha, \psi_\beta \rangle = 0. \tag{3.18}$$

This is a contradiction since the integrand of

$$\Big\langle \psi_\alpha, \prod_{i=1}^{k} (\delta - \delta^{(i)}) \Big\rangle = \int_{\mathcal{D}} \psi_\alpha(\delta) \prod_{i=1}^{k} (\delta - \delta^{(i)}) p(\delta) \, \mathrm{d}\delta \tag{3.19}$$

has a constant sign if ψ_α vanishes only at $\delta^{(1)}$ for $k = 1$. This procedure can be repeated for $k = 2, ..., \alpha - 1$ to show that ψ_α has α distinct real roots in $\mathcal{D}$. □

3.2.4 Function Approximation

As mentioned in Section 3.1, polynomial chaos series expansions promise superior convergence compared to Monte Carlo based methods. Essentially, this is attributed to applying orthogonal polynomials for function approximation, which is the theme of the present section. To set the stage, consider the following well-known result.

Theorem 3.1 (Weierstrass [77, 87])**.** *Let $\mathcal{D} \subset \mathbb{R}$ be a compact set, $f : \mathcal{D} \mapsto \mathbb{R}$ a continuous function, and $\epsilon > 0$. Then there exists a polynomial g such that*

$$\sup_{\delta \in \mathcal{D}} |f(\delta) - g(\delta)| < \epsilon. \tag{3.20}$$

The Weierstrass approximation theorem states that for a given uniform error bound ϵ, there exists a suitable polynomial approximating the function f. Therein, f is solely required to be continuous and have a compact domain. However the, degree of the polynomial may be high and the rate of convergence is not quantified. If, in addition, information on the smoothness of f is given and the approximation error is measured in mean-square with respect to the L^2_p-norm, there exists a stronger result. This is widely known as spectral convergence. In order to elaborate, we first need to introduce the orthogonal projection within the Hilbert space L^2_p as follows.

For $\alpha \in \mathbb{N}_0$, let ψ_α be the orthogonal polynomials with respect to a pdf $p(\delta)$. Denote by $\mathcal{P}_d$ the space of all real polynomials of maximum degree $d \in \mathbb{N}_0$. Then the orthogonal projection of any $f \in L^2_p$ onto $\mathcal{P}_d$ is defined by

$$\Pi_d f := \sum_{\alpha=0}^{d} \frac{\langle f, \psi_\alpha \rangle}{\|\psi_\alpha\|^2_{L^2_p}} \, \psi_\alpha. \tag{3.21}$$

In fact, it can be shown that $\Pi_d f$ is the optimal approximation of f yielding the minimal mean-square error

$$\Pi_d f = \arg\min_{g \in \mathcal{P}_d} \|f - g\|_{L^2_p}, \tag{3.22}$$

see [77]. As a consequence of the orthogonal projection (3.21), the approximation error is of course orthogonal to the projection subspace

$$\langle f - \Pi_d f, h \rangle = 0 \quad \forall h \in \mathcal{P}_d. \tag{3.23}$$

It is highlighted that function approximation as above is completely analogous to the orthogonal projection of a point in the Euclidean space $\mathbb{R}^3$ onto a two-dimensional subspace, i.e., a plane.

Spectral convergence is referred to as the property that the quality of function approximation by orthogonal polynomials improves exponentially as the regularity (i.e., smoothness) of f increases [77]. This is at the core of the convergence behavior of orthogonal series expansions such as the original Fourier series. Note that Equation (3.21) already has the character of such a series expansion. In order to define the notion of spectral convergence in the next theorem, there is one tool remaining to be introduced for characterizing the smoothness of f: Sobolev spaces. Extending the definition of L_p^2, the Sobolev space is the function space

$$H_p^k := \left\{ f \in L_p^2 \;\middle|\; \frac{\partial^j f}{\partial \delta^j} \in L_p^2 \quad \forall j = 0, ..., k \right\} \tag{3.24}$$

equipped with inner product $\langle f, g \rangle_{H_p^k} = \sum_{j=0}^{k} \langle \frac{\partial^j f}{\partial \delta^j}, \frac{\partial^j g}{\partial \delta^j} \rangle_{L_p^2}$ and induced norm $\|f\|_{H_p^k} = \sqrt{\langle f, f \rangle_{H_p^k}}$. Thus the Sobolev space comprises all functions in L_p^2 that are differentiable up to order k, where the respective (weak) derivatives are also in L_p^2. We are now able to state the spectral convergence property of function approximation by Legendre polynomials. For simplicity, we limit the exposition to the orthogonal polynomials with respect to the uniform distribution. The result can be generalized to the multivariate case $n_\delta > 1$ and to the family of all classical orthogonal polynomials with respect to the beta, gamma, and Gaussian distribution, see [2] or [77].

Theorem 3.2 (Spectral convergence of Legendre polynomial expansions [77])**.** *For all $f \in H_p^k$, there exists a constant $C_\pi \geq 0$ such that*

$$\|f - \Pi_d f\|_{L_p^2} := \left\| f - \sum_{\alpha=0}^{d} \frac{\langle f, \psi_\alpha \rangle}{\langle \psi_\alpha^2 \rangle} \psi_\alpha \right\|_{L_p^2} \leq C_\pi d^{-k} \|f\|_{H_p^k}. \tag{3.25}$$

Thus, the smoother the function f, the better the convergence rate of the orthogonal expansion $\Pi_d f$ in mean square. However, discontinuities and poor regularity may result in convergence issues. A classic example is Gibbs' Phenomenon [77]. Luckily, the solutions of the classes of (linear and Riccati) differential equations considered in the scope

of this thesis exhibit certain smoothness and monotonicity properties with respect to the initial data and coefficients. This provides optimism for applying orthogonal polynomial expansions throughout this thesis. Employing orthogonal polynomials for uncertainty quantification of stochastic processes is known as polynomial chaos expansion. The idea is formally introduced in the following section.

3.3 Polynomial Chaos Series Expansion

This section introduces a Fourier series for random variables. The presented material is mainly based on Chapter 11 in [77]. See also [43] for further details.

3.3.1 Wiener Polynomial Chaos

It was originally the observation of Norbert Wiener[3] [90] that any static, i.e., time-invariant, random variable $f \in L_p^2$ can be developed as a generalized Fourier series

$$f(\delta) = \sum_{\alpha=0}^{\infty} f_\alpha \psi_\alpha(\delta). \tag{3.26}$$

Therein, the f_α are deterministic expansion coefficients and ψ_α are the orthogonal polynomials with respect to the probability density function of δ. It is still assumed $n_\delta = 1$. The f_α are often referred to as stochastic modes of f and are determined by standard Hilbert space orthogonal projection, see Equation (3.21)

$$f_\alpha = \frac{\langle f, \psi_\alpha \rangle}{\langle \psi_\alpha, \psi_\alpha \rangle} = \frac{\int_{\mathcal{D}} f(\delta)\psi_\alpha(\delta)p(\delta)\,\mathrm{d}\delta}{\int_{\mathcal{D}} \psi_\alpha^2(\delta)p(\delta)\,\mathrm{d}\delta}. \tag{3.27}$$

In practice, the infinite sum (3.26) needs to be truncated. This means that f is approximated by a finite series expansion

$$f(\delta) \approx \sum_{\alpha=0}^{d} f_\alpha \psi_\alpha(\delta) \tag{3.28}$$

[3] To be precise, Wiener considered only normal distributed random variables δ. This corresponds to orthogonally decomposing L_p^2 by the Hermite polynomials. The expansion series was generalized to orthogonal polynomials of the Askey scheme in [92].

with maximum polynomial degree $d < \infty$. As a consequence of the orthogonal projection, the truncation error is known to be orthogonal to the subspace $\text{span}\{\psi_0, \psi_1, ..., \psi_d\}$ of L_p^2

$$\langle f - \sum_{\alpha=0}^{d} f_\alpha \psi_\alpha,\ \psi_\beta \rangle = \langle \sum_{\alpha>d} f_\alpha \psi_\alpha,\ \psi_\beta \rangle = \sum_{\alpha>d} f_\alpha \langle \psi_\alpha, \psi_\beta \rangle = 0 \quad \forall \beta = 0, ..., d. \tag{3.29}$$

Apart from orthogonality (3.29), the approximation error is known to converge to zero in the mean-square sense, i.e.,

$$\lim_{d\to\infty} \big\| f - \sum_{\alpha=0}^{d} f_\alpha \psi_\alpha \big\|_{L_p^2} = 0. \tag{3.30}$$

This was first established by Cameron and Martin [21]. In addition, according to the spectral convergence theorem (Theorem 3.2) the truncation error is even known to converge exponentially in the L_p^2-norm. The convergence rate accelerates with increasing smoothness, i.e., stochastic regularity, of f. See also the comments after Theorem 3.2.

Another appealing property of polynomial chaos expansions (PCEs) is that the moments can be extracted directly in terms of the expansion coefficients. For instance, the expectation is given simply by the zeroth expansion coefficient

$$\mathbb{E}[f] \approx \mathbb{E}\big[\sum_{\alpha=0}^{d} f_\alpha \psi_\alpha \big] = \langle \sum_{\alpha=0}^{d} f_\alpha \psi_\alpha, \psi_0 \rangle = \sum_{\alpha=0}^{d} f_\alpha \langle \psi_\alpha, \psi_0 \rangle = f_0. \tag{3.31}$$

Similarly, the variance is given by a weighted sum of the squared expansion coefficients

$$\mathbb{V}[f] = \mathbb{E}\big[(f - \mathbb{E}[f])^2\big] \approx \mathbb{E}\big[\big(\sum_{\alpha=0}^{d} f_\alpha \psi_\alpha - f_0 \big)^2\big] = \sum_{\alpha,\beta=1}^{d} f_\alpha f_\beta \langle \psi_\alpha, \psi_\beta \rangle = \sum_{\alpha=1}^{d} f_\alpha^2 \langle \psi_\alpha^2 \rangle. \tag{3.32}$$

3.3.2 Generalized Polynomial Chaos

We now generalize to the multivariate case $n_\delta > 1$. Assume δ is a $\mathbb{R}^{n_\delta}$-valued random vector. Thus, the components δ_i are $i = 1, ..., n_\delta$ independent random variables. The joint probability density function of δ is $p(\delta) = \prod_{i=1}^{n_\delta} p_i(\delta_i)$. Then the corresponding

multivariate orthogonal polynomials can be obtained simply by taking products of the individual univariate orthogonal polynomials. In order to elaborate, for $i = 1, ..., n_\delta$ and $\alpha_i \in \mathbb{N}_0$, denote by $\psi_{\alpha_i}(\delta_i)$ the *univariate* orthogonal basis polynomials with respect to $p_i(\delta_i)$. The associated *multivariate* orthogonal basis polynomials are then given by

$$\psi_\alpha(\delta) = \prod_{i=1}^{n_\delta} \psi_{\alpha_i}(\delta_i), \qquad \alpha_i \in \mathbb{N}_0 \quad i = 1, ..., n_\delta. \tag{3.33}$$

Therein, $\alpha = \alpha_1 \times ... \times \alpha_{n_\delta}$ denotes a n_δ-dimensional multi-index $\alpha \in \mathbb{N}_0^{n_\delta}$. This multi-index notation is standard within the literature, see e.g. [3].

Realistic applications require an approximation within a finite-dimensional polynomial basis. For instance, the space of polynomials with maximum total polynomial degree less than or equal to d is represented by

$$\text{span}\Big\{ \psi_\alpha(\delta) = \prod_{i=1}^{n_\delta} \psi_{\alpha_i}(\delta_i) \quad \Big| \quad |\alpha| = \sum_{i=1}^{n_\delta} \leq d \Big\}. \tag{3.34}$$

This truncation scheme is referred to as total polynomial degree truncation. A finite series expansion as in (3.28) with respect to a multivariate orthogonal polynomial basis is referred to as a generalized polynomial chaos expansion. Selecting a truncation scheme with maximum total polynomial degree d, it can be verified that the generalized PCE

$$f(\delta) \approx \sum_{|\alpha| \leq d} f_\alpha \psi_\alpha(\delta) \tag{3.35}$$

has a number of expansion coefficients given by

$$L = \frac{(n_\delta + d)!}{n_\delta! d!} \tag{3.36}$$

see [77]. Thus the number of expansion terms increases combinatorially with n_δ and d.

Note that several other truncation schemes are used in practice, see e.g. [46]. For instance, for $i = 1, ..., n_\delta$, the polynomial basis can also be truncated such that it has an equal maximum polynomial degree d in each uncertain parameter δ_i. This implies $L = (d + 1)^{n_\delta}$ terms. The polynomial degree may also be refined individually within certain components δ_i of the vector δ.

Expansions of Stochastic Processes

Apart from using the standard multi-index notation, the properties of generalized PCEs remain essentially the same as in the univariate case considered in the previous section. We state these properties here for a stochastic process $y(t,\delta)$ as introduced in the beginning of this chapter (Section 3.1). For such a random mapping

$$y(t,\delta) = \sum_{\alpha \geq 0} y_\alpha(t)\psi_\alpha(\delta) \tag{3.37}$$

the expansion coefficients are still given by the projection relation

$$y_\alpha = \frac{\langle y(t,\cdot), \psi_\alpha \rangle}{\langle \psi_\alpha, \psi_\alpha \rangle} = \frac{\int_{\mathcal{D}} y(t,\delta)\psi_\alpha(\delta)p(\delta)\,\mathrm{d}\delta}{\int_{\mathcal{D}} \psi_\alpha^2(\delta)p(\delta)\,\mathrm{d}\delta}. \tag{3.38}$$

For a finite expansion truncated by a maximum total polynomial degree d, the mean field is approximated by

$$\mathbb{E}[y(t,\cdot)] \approx \mathbb{E}\left[\sum_{|\alpha|\leq d} y_\alpha(t)\psi_\alpha(\delta)\right] = y_0(t). \tag{3.39}$$

Similarly, the covariance can be extracted by

$$\begin{aligned} C(t,t') &= \mathbb{E}\Big[\Big(y(t,\cdot) - \mathbb{E}[y(t,\cdot)]\Big)\Big(y(t',\cdot) - \mathbb{E}[y(t',\cdot)]\Big)^T\Big] \\ &\approx \mathbb{E}\left[\left(\sum_{|\alpha|\leq d} y_\alpha(t)\psi_\alpha - y_0(t)\right)\left(\sum_{|\beta|\leq d} y_\beta(t')\psi_\beta - y_0(t')\right)^T\right] \\ &= \sum_{\substack{\alpha,\beta>0,\\ |\alpha|,|\beta|\leq d}} y_\alpha(t)y_\beta(t')^T\langle\psi_\alpha,\psi_\beta\rangle = \sum_{\substack{\alpha>0,\\ |\alpha|\leq d}} y_\alpha(t)y_\alpha(t')^T\langle\psi_\alpha^2\rangle. \end{aligned} \tag{3.40}$$

The variance can be deduced from Equation (3.40) as

$$\mathbb{V}[y(t,\cdot)] = \mathbb{E}\Big[\Big(y(t,\cdot) - \mathbb{E}[y(t,\cdot)]\Big)\Big(y(t,\cdot) - \mathbb{E}[y(t,\cdot)]\Big)^T\Big] \approx \sum_{\substack{\alpha>0,\\ |\alpha|\leq d}} y_\alpha(t)y_\alpha(t)^T\langle\psi_\alpha^2\rangle. \tag{3.41}$$

Besides the projection relations (3.27), (3.38), we have not yet indicated how to actually compute the series expansion in practice. Specifically for the case of stochastic

processes $y(t, \delta)$ considered within this monograph (see Section 3.1), the only question remaining is how to compute the expansion coefficients of the solution. This is the theme of the following two sections.

Two approaches are generally distinguished: intrusive (Section 3.4) and non-intrusive (Section 3.5) methods. Non-intrusive methods consider the model (2.1) as a black-box and approximate the projection integrals numerically using realizations of δ. For example, deterministic Gaussian quadrature, Monte Carlo (MC) sampling, or quasi-MC sampling may be applied. Intrusive methods, on the other hand, build a generally coupled system of equations of increased dimension for the expansion coefficients. In particular, the Galerkin approach introduced in the following section projects the original model equations on the considered orthogonal polynomial basis. Intrusive methods preserve the formal mathematical structure of the considered random mapping [e.g. $f(\delta)$ or (2.1)]. This means that, for instance, the series expansion of a random parameter-dependent Riccati differential equation (RDE) remains a RDE, albeit generally a coupled *system* of RDEs.

3.4 Galerkin Projection

We start this section with an introductory example. The general method is outlined in Section 3.4.2.

3.4.1 Introductory Example

Consider the following uncertain linear time-invariant system

$$\dot{x}(t, \delta) = A(\delta)x(t, \delta) \tag{3.42}$$

with $A : \mathcal{D} \mapsto \mathbb{R}^{n_x \times n_x}$. Applying Galerkin projection based PCE to (3.42) and ordinary differential equations in general consists of the following steps:

1. Insert the series expansion for the input uncertainty δ and the model output, i.e., $x(t, \delta)$ in (3.42).

2. Multiply the model equation by each ψ_α.

3. Compute the projection integral $\langle \cdot, \psi_\alpha \rangle$.

4. Solve the Galerkin projected system of equations for the expansion coefficients.

Here ψ_α are the orthogonal polynomials with respect to the pdf of δ. A key point when applying PCEs for uncertainty quantification is that the model output is projected on the same polynomial basis as the input uncertainty δ, i.e., in case of the system (3.42)

$$x(t, \delta) = \sum_{\alpha \geq 0} x_\alpha(t) \psi_\alpha(\delta). \tag{3.43}$$

Therein, the coefficients $x_\alpha : [0, \infty) \mapsto \mathbb{R}^{n_x}$ are defined by the standard projection relation, evaluated component-wise

$$x_{i,\alpha}(t) = \frac{1}{\langle \psi_\alpha^2 \rangle} \langle x_i(t, \cdot), \psi_\alpha \rangle \quad \text{for } i = 1, ..., n_x. \tag{3.44}$$

Inserting (3.43) into (3.42) and projecting onto each ψ_α yields

$$\begin{aligned} \left\langle \psi_\alpha, \sum_{\beta \geq 0} \dot{x}_\beta(t) \psi_\beta \right\rangle &= \left\langle \psi_\alpha, \sum_{\beta \geq 0} A x_\beta(t) \psi_\beta \right\rangle \\ \dot{x}_\alpha(t) \langle \psi_\alpha^2 \rangle &= \sum_{\beta \geq 0} \langle \psi_\alpha A \psi_\beta \rangle x_\beta(t) \\ \dot{x}_\alpha(t) &= \frac{1}{\langle \psi_\alpha^2 \rangle} \sum_{\beta \geq 0} \langle \psi_\alpha A \psi_\beta \rangle x_\beta(t). \end{aligned} \tag{3.45}$$

Also here the projection integrals are evaluated component-wise. Thus, the expansion coefficients $x_\alpha(t)$ are governed by a deterministic, coupled system (3.45) of linear differential equations. Of course, practical applications demand a finite series expansion. For instance, if the PCE is truncated at a maximum total polynomial degree d, the sums in (3.45) have $L = \frac{(n_\delta + d)!}{n_\delta! d!}$ terms, see Equation (3.36). The coupled system (3.45) can then be written in matrix-vector form

$$\dot{X}(t) = \mathcal{A} X(t) \tag{3.46}$$

where $X : [0, \infty) \mapsto \mathbb{R}^{n_x L}$ concatenates vertically the expansion coefficients $x_\alpha(t)$. $\mathcal{A} \in \mathbb{R}^{(n_x L) \times (n_x L)}$ is a block matrix where the (α, β)th block is given by

$\mathcal{A}_{\alpha\beta} = \frac{1}{\langle\psi_\alpha^2\rangle}\langle\psi_\alpha A\psi_\beta\rangle$. Herein, $\alpha, \beta \in \mathbb{N}_0$ are used as scalar indices. Switching between multi-index notation $\alpha \in \mathbb{N}_0^{n_\delta}$ and scalar indices is straightforward by ordering the polynomial basis, e.g., lexicographically, see [77]. In any case, Greek letters (α, β, ...) are used to index within the polynomial basis, whereas Latin characters (i, j, ...) index spatially into matrices and vectors, see for example Equation (3.44).

The exemplary projection of the system (3.42) unveils a recurring feature of Galerkin projection: the governing system of equations for the stochastic modes (3.45) or (3.46) has a similar mathematical structure as the original system (3.42). However, the expansion coefficients are generally coupled together by a system of enhanced dimension, i.e., increased by the factor L. This implies that in general, the existing numerical solver for the original problem may need to be adapted to account for the coupled problem. This is in contrast to the methods considered in the subsequent section. Non-intrusive approaches calculate the stochastic modes by using realizations of the original system. Thus, the existing numerical solver can be reused and the governing system equations remain unaltered [77].

Example 3.3. *The truncated polynomial chaos expansion with total degree $d \leq 5$ for*

$$A(\delta) = \begin{bmatrix} 0 & 1 \\ -1.8 + \delta_1 & -1.4 + \delta_2 \end{bmatrix} \tag{3.47}$$

and uncertain initial condition $x(0, \delta) = [0.75 + \delta_3, 0]^T$ is plotted in Figure 3.4. All $n_\delta = 3$ parameters are distributed uniformly, i.e., $\delta_1 \sim \mathcal{U}(0, 1.6)$, $\delta_2 \sim \mathcal{U}(0, 1.2)$, and $\delta_3 \sim \mathcal{U}(0, 0.5)$. Thus the respective orthogonal polynomials are the Legendre polynomials. Mean and standard deviation of the PCE approximation are plotted for the first coordinate $x_1(t, \delta)$ of the state vector. The approximation agrees well and converges rapidly to the exact quantities obtained by high-order numerical integration (indicated in red). The first few stochastic modes of the expansion are also plotted.

3.4.2 General Approach

Applying Galerkin projection to random ordinary differential equations with uncertain initial conditions, coefficients, etc., is essentially analogous to the example given above, see [43] for detailed applications. The general idea is given in [77] as follows. Denote the

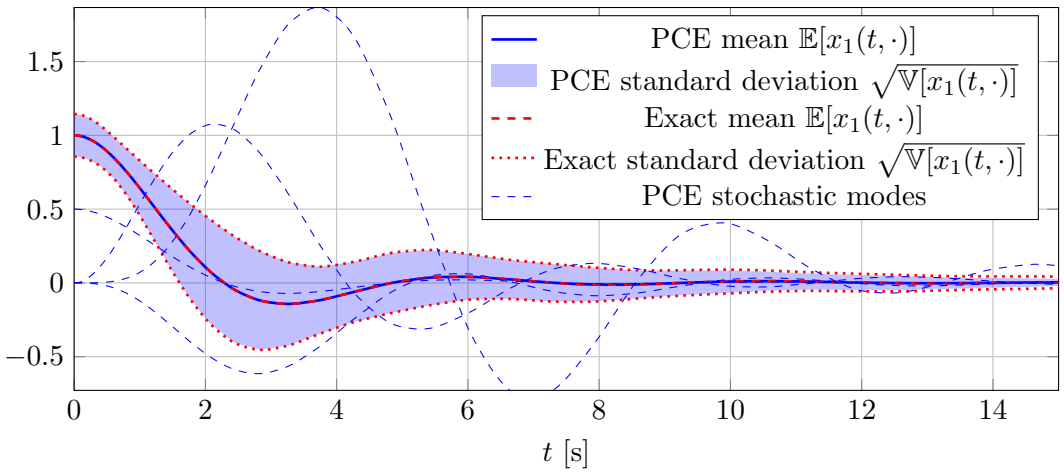


Figure 3.4: Galerkin Projection Example

model relationship between input uncertainty δ and output solution y by some formal equation

$$\mathcal{F}(y;\delta) = 0. \tag{3.48}$$

This may be an ordinary differential equation, i.e., the setting considered in Figure 3.1; generally even a (set of) algebraic or partial differential equation(s). Instead of solving the Equation (3.48) for all $\delta \in \mathcal{D}$ — which may be computationally impossible in practice — the Galerkin method, also known as Ritz-Galerkin approach, searches for "weak" solutions, determined by inner products

$$\langle \psi_\alpha,\ \mathcal{F}(y;\delta)\rangle = 0 \quad \forall \alpha = 0, ..., L-1. \tag{3.49}$$

This weak interpretation is more appealing both in theory and practice than the original hard problem (3.48). Generally, the softer requirement that (3.49) must hold within a finite-dimensional subspace is at the core of countless numerical algorithms.

It is highlighted that the cornerstone within polynomial chaos theory lies in applying the same orthogonal polynomial basis for the expansion of the output solution as for the input uncertainty. This is argued in [43] to be attributed to the fact that, for instance, if the solution y is distributed normally, the Hermite polynomials constitute

the optimal basis since any arbitrarily normal distributed $y \sim \mathcal{N}(\mu, \sigma^2)$ can be represented exactly by the *standard* normal distribution $\delta \sim \mathcal{N}(0,1)$: $y = \mu + \sigma\delta$, i.e., a first-order series expansion. This can be generalized to the assertion that the optimal orthogonal polynomial basis is of course given by the probability law of the solution we seek to quantify. However, apart perhaps from linear systems excited by Gaussian white noise, this setting is actually seldom encountered in practice. In realistic applications, especially in case of complex model-based predictions, one does not know a-priori the distribution of the solution y. The hope is, since one knows the optimal polynomial basis for the input uncertainty δ, the same expansion basis will work well for the model output y. The spectral convergence theorem (Theorem 3.2) makes this leap of faith small for smooth problems.

3.5 Numerical integration

Galerkin projection is referred to as an intrusive approach since the governing system equations are "intruded", i.e., projected on a finite-dimensional basis, to form a deterministic system of coupled equations for the expansion coefficients. In contrast, "non-intrusive" methods rely on realizations to approximate the expansion coefficients defined by the projection integrals (3.38). Therefore, the original simulation code or numerical solver can be used as-is, i.e., it is treated as a black-box. All that is required is to be able to solve for the model output given a realization of δ. In essence, non-intrusive methods can be thought of as using realizations of δ within a numerical integration scheme.

Note that intrusive (Galerkin projection) and non-intrusive methods may also be distinguished by the order in which the original formal problem solver and numerical integration, i.e., evaluation of the projection integrals, are applied. In practice the two approaches yield different approximations. For instance, Galerkin projection of an uncertain differential equation is generally not equivalent to solving for all δ and projecting afterwards (see Section 4.5.5).

3.5.1 Gaussian Quadrature

One possibility from the field of numerical integration is to apply a deterministic quadrature rule in order to approximate the projection integrals (3.38). Quadrature means numerical approximation of an integral by a finite sum, i.e.,

$$\mathbb{E}[f] = \int_{\mathcal{D}} f(\delta)p(\delta)\,\mathrm{d}\delta \approx \sum_{i=1}^{N} w_i f(\delta^{(i)}). \tag{3.50}$$

A quadrature rule determines its nodes $\delta^{(i)} \in \mathcal{D}$ and scalar weights $w_i \in \mathbb{R}$ for $i = 1, ..., N$ solely based on the probability density $p(\delta)$ and independent of the integrand $f(\delta)$, see [77].

For instance, the method of Gaussian quadrature approximates the expansion coefficients (3.38) by

$$y_\alpha(t)\langle\psi_\alpha^2\rangle = \langle y(t,\cdot), \psi_\alpha\rangle \approx \sum_{i=1}^{N} w_i\ y(t,\delta^{(i)})\psi_\alpha(\delta^{(i)}). \tag{3.51}$$

Therein, $\delta^{(i)}$ are prescribed realizations of $\delta \in \mathcal{D}$. These are given together with the associated weights in the following definition, taken from [77].

Definition 3.2. *Let ψ_N, $N \in \mathbb{N}$, be the Nth orthogonal polynomial with respect to $p(\delta)$. The N-point Gauss quadrature rule is then determined by nodes (also known as Gauss points) given by the roots of ψ_N. The associated weights are*

$$w_i = \int_{\mathcal{D}} \prod_{\substack{1\le j\le n,\\ j\neq i}} \frac{\delta - \delta^{(j)}}{\delta^{(i)} - \delta^{(j)}} p(\delta)\,\mathrm{d}\delta. \tag{3.52}$$

The roots of orthogonal polynomials are investigated in Section 3.2.3. We highlight here that the N-point Gauss quadrature rule *exactly* integrates polynomials of degree $2N-1$. For instance, the normalization constants $\langle\psi_\alpha^2\rangle$ required by the projection relation (3.51) can be determined exactly by Gaussian quadrature of order $N \geq (\alpha+1)/2$. It is noted further that the N-point Gauss quadrature rule is *optimal* for polynomial integrands in the sense that there exists no other quadrature formula with N nodes and higher order of accuracy. The reader is referred to [77] for details.

3.5.2 Monte Carlo Methods

In contrast to deterministic quadrature, where the nodes are fixed by the probability density $p(\delta)$, random sampling-based methods, i.e., Monte Carlo, and pseudo-random methods, i.e., quasi-Monte Carlo, may also be employed. These methods randomly generate samples distributed according to $p(\delta)$ in order to numerically approximate the projection integrals

$$y_\alpha(t)\langle\psi_\alpha^2\rangle = \langle y(t,\cdot),\psi_\alpha\rangle \approx \frac{1}{N}\sum_{i=1}^{N} y(t,\delta^{(i)})\psi_\alpha(\delta^{(i)}). \tag{3.53}$$

For instance, a classical Monte Carlo estimator of the expectation is obtained for $\alpha = 0$ (since $\psi_0 = 1$), i.e., the weights in Equation (3.50) are set to $w_i = 1/N$.

There are no answers, only cross references.

Norbert Wiener

Chapter 4

LFT Projection

This chapter presents a chief theoretical and practical contribution of this thesis. The problem setting is illustrated in Figure 4.1 — what happens when we apply the polynomial chaos series expansion introduced in Chapter 3 to an uncertain system in linear fractional representation? The answer to this question is two-fold.

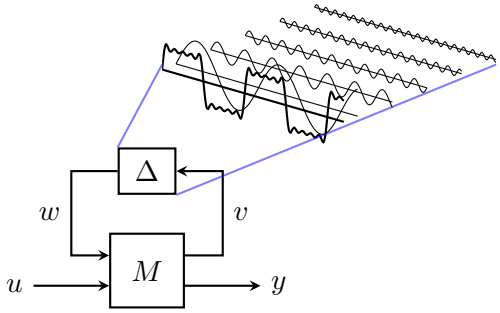


Figure 4.1: Generalized Fourier Series Expansion of a LFT

First of all, the Galerkin projection of LFTs (see Section 4.3) leads to significant computational benefits compared to the conventional Galerkin approach (Section 4.2). It is illustrated by means of an example in Section 4.4 that employing a LFT substantially relieves the Galerkin projection of work.

Ultimately, applying LFT Galerkin projection leads to a "cross reference". It is shown in Section 4.5 that — ironically — intrusive Galerkin projection of uncertain systems in linear fractional representation is equivalent to the non-intrusive approach: simply applying Gaussian quadrature.

4.1 Kronecker Product Notation

In order to clearly and concisely present the arguments throughout this chapter, we briefly introduce the notation for Kronecker products in this section. Given two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$, the *Kronecker product* of A and B is

$$A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{bmatrix} \in \mathbb{R}^{mp\times nq}. \tag{4.1}$$

Some of its main properties are

1. $A \otimes 1 = A = 1 \otimes A$
2. $(A \otimes B) \otimes C = A \otimes (B \otimes C)$
3. $(A + B) \otimes C = (A \otimes C) + (B \otimes C)$
4. $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$
5. $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$.

Now let the two matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{m\times m}$ be square. Denote by λ_i, v_i the $i = 1, ..., n$ eigenvalues and eigenvectors of A such that $Av_i = \lambda_i v_i$. Similarly, let μ_j and w_j denote the $j = 1, ..., m$ eigenvalues and eigenvectors of B (i.e., $Bw_j = \mu_j w_j$). Then the following property adds to our list.

6. The eigenvalues of $A \otimes B$ are the nm numbers $\lambda_i \mu_j$ for $i = 1, ..., n$ and $j = 1, ..., m$. The corresponding eigenvectors are $v_i \otimes w_j$.

4.2 Galerkin Projection based PCE of Linear Systems

Within this section, we focus on the conventional approach of Galerkin projection based polynomial chaos applied to uncertain linear systems, see for instance [29] or [73]. In the following, we consider the general case of uncertain LTV systems. The application to uncertain LTI systems follows simply as a special case.

Recall that applying Galerkin projection based PCE according to Section 3.4 roughly breaks down to inserting the series expansion for all random quantities and computing the scalar product with respect to the basis functions $\psi_\alpha(\delta)$. For an uncertain LTV system (2.6) the random variables[1] are $x(t,\delta)$ and $y(t,\delta)$. These are to be approximated by truncated series expansions

$$x(t,\delta) \approx \sum_{\alpha=0}^{L-1} x_\alpha(t)\psi_\alpha(\delta), \qquad y(t,\delta) \approx \sum_{\alpha=0}^{L-1} y_\alpha(t)\psi_\alpha(\delta) \tag{4.2}$$

with respect to a suitable orthogonal polynomial basis of δ and some finite $L < \infty$. Therein, the deterministic expansion coefficients are formally defined by the usual projection relations

$$x_\alpha(t) = \frac{1}{\langle\psi_\alpha^2\rangle}\langle x(t,\cdot),\psi_\alpha\rangle, \qquad y_\alpha(t) = \frac{1}{\langle\psi_\alpha^2\rangle}\langle y(t,\cdot),\psi_\alpha\rangle. \tag{4.3}$$

Now denote by

$$\Psi(\delta) := \big[\psi_0(\delta),\dots,\psi_{L-1}(\delta)\big]^T \in \mathbb{R}^L \tag{4.4}$$

the vector concatenating vertically the polynomial basis functions ψ_α for $\alpha = 0,...,L-1$. The projection relations (4.3) may then be summarized conveniently $\forall\alpha = 0,...,L-1$ using Kronecker product notation:

$$\begin{aligned} X(t) &= \big(I_{n_x} \otimes \langle\Psi\Psi^T\rangle\big)^{-1}\langle x(t,\cdot)\otimes\Psi\rangle \\ Y(t) &= \big(I_{n_y} \otimes \langle\Psi\Psi^T\rangle\big)^{-1}\langle y(t,\cdot)\otimes\Psi\rangle. \end{aligned} \tag{4.5}$$

Herein, the capital vectors $X(t) \in \mathbb{R}^{n_x L}$, $Y(t) \in \mathbb{R}^{n_y L}$ concatenate the expansion coefficients (4.3) of the lower-case vectors $x(t,\delta)$, $y(t,\delta)$. Note that $\langle\Psi\Psi^T\rangle$ is simply a diagonal matrix with $\mathbb{E}[\psi_\alpha^2]$ on the main diagonal.

The series expansion (4.2) may be written as

$$x(t,\delta) \approx \big[I_{n_x} \otimes \Psi^T(\delta)\big]X(t), \qquad y(t,\delta) \approx \big[I_{n_y} \otimes \Psi^T(\delta)\big]Y(t). \tag{4.6}$$

[1]Strictly speaking $x(t,\delta)$ and $y(t,\delta)$ are stochastic processes. Also δ is a random vector and its expansion $\delta = \sum_{\alpha=0}^{L-1}\delta_\alpha\psi_\alpha(\delta)$ is inserted as well. Often, the expansion of δ is exact for a first-order series expansion, e.g., with respect to the classical orthogonal polynomials. It is therefore omitted from the main text for brevity.

Galerkin projection is then applied as follows. Insert the series expansion (4.6) into the LTV system dynamics (2.6) to obtain

$$\begin{aligned}\big[I_{n_x}\otimes\Psi^T(\delta)\big]\dot{X}(t) &= A(t,\delta)\big[I_{n_x}\otimes\Psi^T(\delta)\big]X(t)+B(t,\delta)u(t)\\ \big[I_{n_y}\otimes\Psi^T(\delta)\big]Y(t) &= C(t,\delta)\big[I_{n_x}\otimes\Psi^T(\delta)\big]X(t)+D(t,\delta)u(t).\end{aligned} \tag{4.7}$$

Project Equation (4.7) onto each ψ_α for $\alpha = 0,...,L-1$ using Kronecker product notation, i.e., $\langle(\cdot)\otimes\Psi\rangle$, yielding

$$\begin{aligned}\big\langle\big([I_{n_x}\otimes\Psi^T]\dot{X}\big)\otimes\Psi\big\rangle &= \big\langle\big(A[I_{n_x}\otimes\Psi^T]X\big)\otimes\Psi\big\rangle+\big\langle\big(Bu\big)\otimes\Psi\big\rangle\\ \big\langle\big([I_{n_y}\otimes\Psi^T]Y\big)\otimes\Psi\big\rangle &= \big\langle\big(C[I_{n_x}\otimes\Psi^T]X\big)\otimes\Psi\big\rangle+\big\langle\big(Du\big)\otimes\Psi\big\rangle,\end{aligned} \tag{4.8}$$

where we have omitted function arguments in the interest of clarity. Using properties 1, 2, and 4 of the Kronecker product together with $\Psi = \Psi\cdot 1$, Equation (4.8) can be simplified to

$$\begin{aligned}\langle I_{n_x}\otimes\Psi^T\otimes\Psi\rangle\dot{X}(t) &= \langle A(t,\cdot)\otimes(\Psi\Psi^T)\rangle X(t)+\langle B(t,\cdot)\otimes\Psi\rangle u(t)\\ \langle I_{n_y}\otimes\Psi^T\otimes\Psi\rangle Y(t) &= \langle C(t,\cdot)\otimes(\Psi\Psi^T)\rangle X(t)+\langle D(t,\cdot)\otimes\Psi\rangle u(t).\end{aligned} \tag{4.9}$$

Since $\Psi^T\otimes\Psi = \Psi\Psi^T$, we can multiply Equation (4.9) from the left by $(I\otimes\langle\Psi\Psi^T\rangle)^{-1}$

$$\begin{aligned}\dot{X}(t) &= \underbrace{\big(I_{n_x}\otimes\langle\Psi\Psi^T\rangle\big)^{-1}\big\langle A(t,\cdot)\otimes(\Psi\Psi^T)\big\rangle}_{=:\mathcal{A}(t)}X(t)+\underbrace{\big(I_{n_x}\otimes\langle\Psi\Psi^T\rangle\big)^{-1}\big\langle B(t,\cdot)\otimes\Psi\big\rangle}_{=:\mathcal{B}(t)}u(t)\\ Y(t) &= \underbrace{\big(I_{n_y}\otimes\langle\Psi\Psi^T\rangle\big)^{-1}\big\langle C(t,\cdot)\otimes(\Psi\Psi^T)\big\rangle}_{=:\mathcal{C}(t)}X(t)+\underbrace{\big(I_{n_y}\otimes\langle\Psi\Psi^T\rangle\big)^{-1}\big\langle D(t,\cdot)\otimes\Psi\big\rangle}_{=:\mathcal{D}(t)}u(t).\end{aligned} \tag{4.10}$$

This is a deterministic (i.e., certain) LTV system of dimension increased by a factor of L, i.e., $\mathcal{A}(t)\in\mathbb{R}^{(n_xL)\times(n_xL)}$, $\mathcal{B}(t)\in\mathbb{R}^{(n_xL)\times n_u}$, $\mathcal{C}(t)\in\mathbb{R}^{(n_yL)\times(n_xL)}$, and $\mathcal{D}(t)\in\mathbb{R}^{(n_yL)\times n_u}$. Therein, the expansion coefficients are generally coupled together, e.g., $\mathcal{A}(t)$ is a full matrix. A possibly uncertain initial condition may be projected according to Equation (4.5):

$$X(0) = \big(I_{n_x}\otimes\langle\Psi\Psi^T\rangle\big)^{-1}\langle x(0,\cdot)\otimes\Psi\rangle. \tag{4.11}$$

Note that the Galerkin projected system (4.10) only approximates the exact expansion coefficients formally defined by (4.3), (4.5) starting from the projection of the initial condition (4.11). In general, there will be an error — due to truncation — between the exact coefficients (4.5) and the finite-dimensional approximation given by the Galerkin projected system (4.10).

In contrast, the required projection integrals in Equation (4.10) are assumed to be computable exactly, for instance, by Gaussian quadrature. Based on the dependency on δ, this can be a challenge. See for example [43] for a detailed exposition of how to tackle common nonlinearities. As a rough illustration, consider a system matrix $A(t, \delta)$ depending on products of independent random variables, with total polynomial degree d. In order to evaluate the projection integrals $\langle A(t, \cdot) \otimes (\Psi\Psi^T) \rangle$ exactly, it is necessary to compute a multiplication tensor [43, 25] of order $d + 2$. Generally, this multiplication tensor has L^{d+2} entries, such that storage and computation quickly becomes computationally expensive [77].

If, however, the uncertain LTV system can be written as a linear fractional transformation — which is always possible for rational parameter-dependencies on δ — Galerkin projection can be applied to the system in linear fractional representation, see Figure 4.1. Throughout the thesis, we refer to this approach as LFT Galerkin projection. LFT Galerkin projection may offer substantial computational benefits. The idea, originally published in [25], is presented in the subsequent section.

4.3 LFT Galerkin Projection

Assume the uncertain LTV system (2.6) can be written as a linear fractional transformation (see Section 2.4). The deterministic nominal part is

$$\begin{bmatrix} \dot{x}(t,\delta) \\ v(t,\delta) \\ y(t,\delta) \end{bmatrix} = \begin{bmatrix} A_M(t) & B_w(t) & B_u(t) \\ C_v(t) & D_{vw}(t) & D_{vu}(t) \\ C_y(t) & D_{yw}(t) & D_{yu}(t) \end{bmatrix} \begin{bmatrix} x(t,\delta) \\ w(t,\delta) \\ u(t) \end{bmatrix} \tag{4.12}$$

and the isolated uncertainty enters via the *affine* mapping

$$w(t,\delta) = \Delta(\delta)v(t,\delta), \qquad \Delta(\delta) = \begin{bmatrix} \delta_1 I_{m_1} & & 0 \\ & \ddots & \\ 0 & & \delta_{n_\delta} I_{m_{n_\delta}} \end{bmatrix} \in \mathbb{R}^{n_\Delta \times n_\Delta}. \tag{4.13}$$

In order to determine the expansion coefficients, the idea is now to apply Galerkin projection to the uncertain system in linear fractional representation, as indicated in Figure 4.1.

The procedure is analogous to the derivation presented in the previous section. In addition to the expansion (4.6) of $x(t,\delta)$ and $y(t,\delta)$, the PCE is also applied to $w(t,\delta)$ and $v(t,\delta)$ within the LFT model (4.12), (4.13):

$$w(t,\delta) \approx \big[I_{n_\Delta} \otimes \Psi^T(\delta)\big] W(t), \quad v(t,\delta) \approx \big[I_{n_\Delta} \otimes \Psi^T(\delta)\big] V(t). \tag{4.14}$$

Similar to Equation (4.5), the expansion coefficients of $w(t,\delta)$ and $v(t,\delta)$ are concatenated in capital vectors $W(t), V(t) \in \mathbb{R}^{n_\Delta L}$

$$W(t) = \big(I_{n_\Delta} \otimes \langle \Psi\Psi^T \rangle\big)^{-1} \langle w(t,\cdot) \otimes \Psi \rangle, \quad V(t) = \big(I_{n_\Delta} \otimes \langle \Psi\Psi^T \rangle\big)^{-1} \langle v(t,\cdot) \otimes \Psi \rangle. \tag{4.15}$$

Inserting the series expansion for x, y, w, and v into Equation (4.12) and projecting with $\langle (\cdot) \otimes \Psi \rangle$ yields

$$\left\langle \begin{bmatrix} (I_{n_x} \otimes \Psi^T)\dot{X}(t) \\ (I_{n_\Delta} \otimes \Psi^T)V(t) \\ (I_{n_y} \otimes \Psi^T)Y(t) \end{bmatrix} \otimes \Psi \right\rangle = \left\langle \left\{ \begin{bmatrix} A_M(t) & B_w(t) & B_u(t) \\ C_v(t) & D_{vw}(t) & D_{vu}(t) \\ C_y(t) & D_{yw}(t) & D_{yu}(t) \end{bmatrix} \begin{bmatrix} (I_{n_x} \otimes \Psi^T)X(t) \\ (I_{n_\Delta} \otimes \Psi^T)W(t) \\ u(t) \end{bmatrix} \right\} \otimes \Psi \right\rangle. \tag{4.16}$$

Using the properties of the Kronecker product, it is straightforward to show Equation (4.16) can be simplified to

$$\begin{bmatrix} \dot{X}(t) \\ V(t) \\ Y(t) \end{bmatrix} = \begin{bmatrix} A_M(t) \otimes I_L & B_w(t) \otimes I_L & B_u(t) \otimes e_1 \\ C_v(t) \otimes I_L & D_{vw}(t) \otimes I_L & D_{vu}(t) \otimes e_1 \\ C_y(t) \otimes I_L & D_{yw}(t) \otimes I_L & D_{yu}(t) \otimes e_1 \end{bmatrix} \begin{bmatrix} X(t) \\ W(t) \\ u(t) \end{bmatrix} \tag{4.17}$$

where $e_1 = \langle \Psi \rangle = [1, 0, ..., 0]^T \in \mathbb{R}^L$ is the L-dimensional unit vector. Galerkin projection of the isolated uncertain mapping (4.13) leads to

$$\begin{aligned} W(t) &= \left(I_{n_\Delta} \otimes \langle \Psi \Psi^T \rangle\right)^{-1} \langle \Delta \otimes (\Psi \Psi^T) \rangle V(t) \\ &=: \Delta_\Pi V(t). \end{aligned} \tag{4.18}$$

The projection of the matrix Δ is denoted by $\Delta_\Pi \in \mathbb{R}^{(n_\Delta L)\times(n_\Delta L)}$. Since Δ is a diagonal matrix, Δ_Π is a block-diagonal matrix

$$\Delta_\Pi = \begin{bmatrix} I_{m_1} \otimes (\langle \Psi \Psi^T \rangle^{-1} \langle \delta_1 \Psi \Psi^T \rangle) & & 0 \\ & \ddots & \\ 0 & & I_{m_{n_\delta}} \otimes (\langle \Psi \Psi^T \rangle^{-1} \langle \delta_{n_\delta} \Psi \Psi^T \rangle) \end{bmatrix}. \tag{4.19}$$

It is highlighted that since the nominal part (4.12) is purely deterministic, the stochastic modes in the Galerkin projection (4.17) are *decoupled* from each other. This has to be contrasted to the conventional Galerkin projection (4.10) applied to the original uncertain LTV system, which generally leads to *mode coupling*, i.e., a fully coupled, deterministic LTV system of increased dimension for the expansion coefficients.

In general, Galerkin projection of rational parameter-dependent LTV systems may be a challenge, with computational cost increasing with the complexity of the nonlinear parameter-dependence, see [43]. For instance, it may be required to compute high-order multiplication tensors. Since $\Delta(\delta)$ depends linearly on δ, coupling of the stochastic modes is reduced and isolated by the LFT to the simple multiplication (4.13) [or (4.18)]. In Equation (4.18), solely the third-order multiplication tensor $T_{\alpha\beta\xi} = \langle \psi_\alpha \psi_\beta \psi_\xi \rangle / \langle \psi_\xi \psi_\xi \rangle$ is required, independent of how δ originally enters the uncertain LTV system. Intuitively, this relieves the Galerkin projection of work by separating the system in deterministic and stochastic parts a priori. In contrast, the conventional Galerkin approach projects the system as is.

Furthermore, the third-order multiplication tensor $T_{\alpha\beta\xi}$ has a particularly advantageous, sparse structure since Δ depends affinely on δ. Hence, Δ has an exact first-order expansion, and Δ_Π is a sparse matrix as well. The sparsity pattern of Δ_Π is displayed in Figure 4.2 exemplarily for $n_\delta = 3$ independent uniform distributed random variables δ_i, $i \in \{1, 2, 3\}$. The polynomial basis is constructed using total degree truncation. For

increasing total polynomial degrees d, the markers indicate the nonzero entries of Δ_Π, assuming each δ_i is not repeated, i.e., the multiplicity is $m_i = 1$.

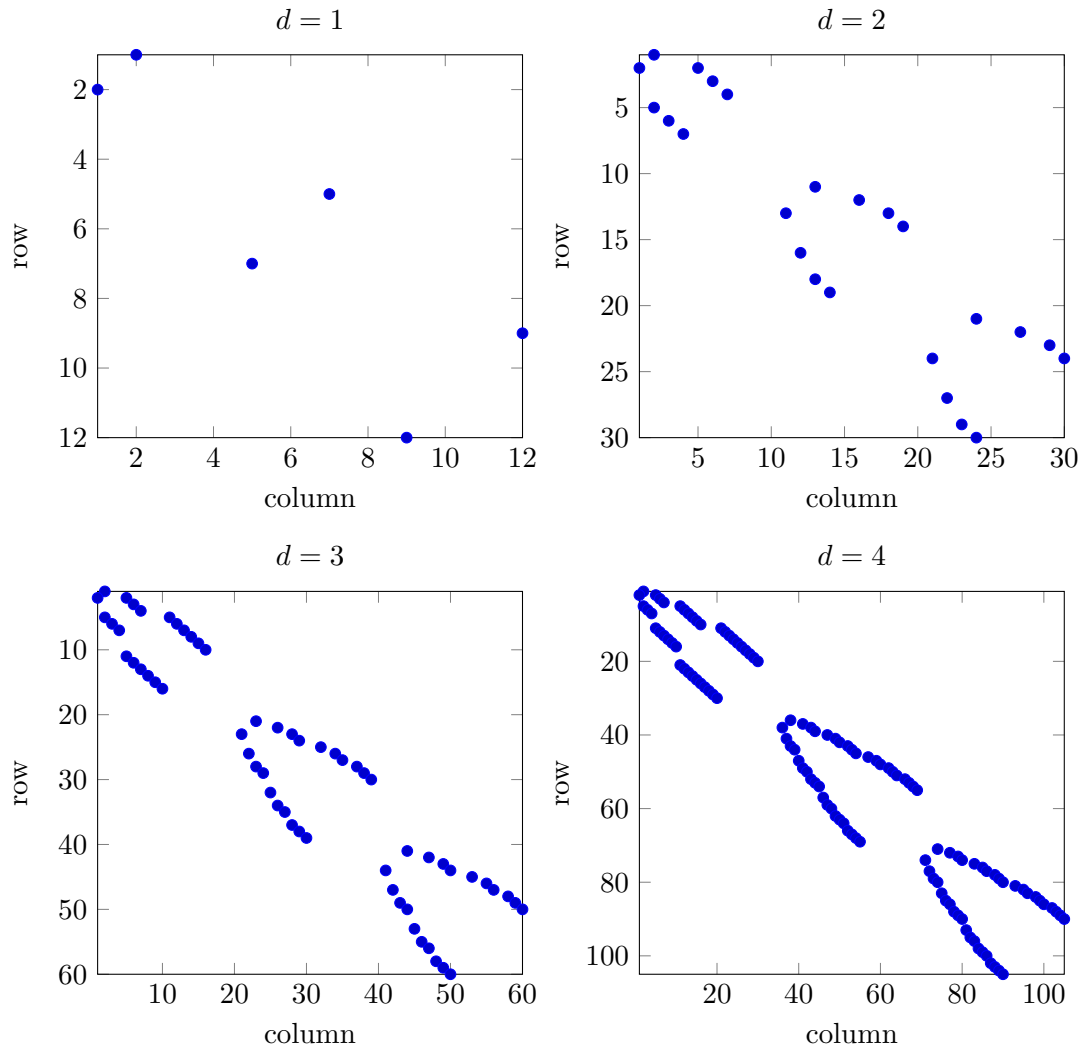


Figure 4.2: Sparsity of Δ_Π for $n_\delta = 3$, increasing total polynomial degrees d, and constant multiplicity $m_i = 1$ for $i \in \{1, 2, 3\}$.

The decoupling and simplicity of Galerkin projection based PCE applied to uncertain systems in LFT form can be exploited both theoretically and numerically. This offers substantial advantages compared to the conventional Galerkin approach indicated

in the previous section, where treating rational parameter-dependencies may be computationally expensive. We illustrate this by means of an example in the following section.

4.4 Two-link Robot Example

The computational benefits of LFT Galerkin projection are now demonstrated by means of an application example. This section is presented originally in the author's work [25].

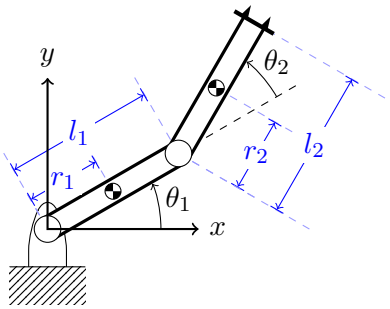


Figure 4.3: Two-Link Planar Manipulator [25]

Consider the two-link robotic manipulator depicted in Figure 4.3. The governing rigid-body equations of motion can be derived by means of the Lagrange formalism [52]

$$\begin{bmatrix} p_1+2p_2\cos(\theta_2) & p_3+p_2\cos(\theta_2) \\ p_3+p_2\cos(\theta_2) & p_3 \end{bmatrix}\begin{bmatrix}\ddot{\theta}_1 \\ \ddot{\theta}_2\end{bmatrix}+\begin{bmatrix} -p_2\sin(\theta_2)\dot{\theta}_2 & -p_2\sin(\theta_2)(\dot{\theta}_1+\dot{\theta}_2) \\ p_2\sin(\theta_2)\dot{\theta}_1 & 0 \end{bmatrix}\begin{bmatrix}\dot{\theta}_1 \\ \dot{\theta}_2\end{bmatrix}=\begin{bmatrix}\tau_1 \\ \tau_2\end{bmatrix}. \tag{4.20}$$

The first summand comprises inertial forces, the second summand the Coriolis and centrifugal forces. The torques applied at the system joints are denoted by $\tau = [\tau_1, \tau_2]^T$. Mass and inertia parameters of the robot are collected in

$$\begin{aligned} p_1 &= I_1 + I_2 + m_1 r_1^2 + m_2(l_1^2 + r_2^2) \\ p_2 &= m_2 l_1 r_2 \\ p_3 &= I_2 + m_2 r_2^2. \end{aligned} \tag{4.21}$$

Here, for $i \in \{1, 2\}$, I_i is the moment of inertia, m_i the mass, and l_i the length of each link. As indicated in Figure 4.3, r_i denotes the distance between the ith center of mass

and joint.

A desired trajectory for the tool center point (TCP) over a finite time horizon $T = 5$ s is displayed in Figure 4.4. Denote by $\Theta := [\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2]^T$. A nominal reference trajectory $\Theta^*(t)$, $\dot{\Theta}^*(t)$, and $\tau^*(t)$ for $t \in [0, T]$ can be deduced from the desired TCP trajectory via inverse kinematics [52]. It is then straightforward to derive a nominal LTV model by linearizing with respect to the reference trajectory. This may be accomplished analytically or, e.g., by automatic differentiation within the Julia programming language [7].

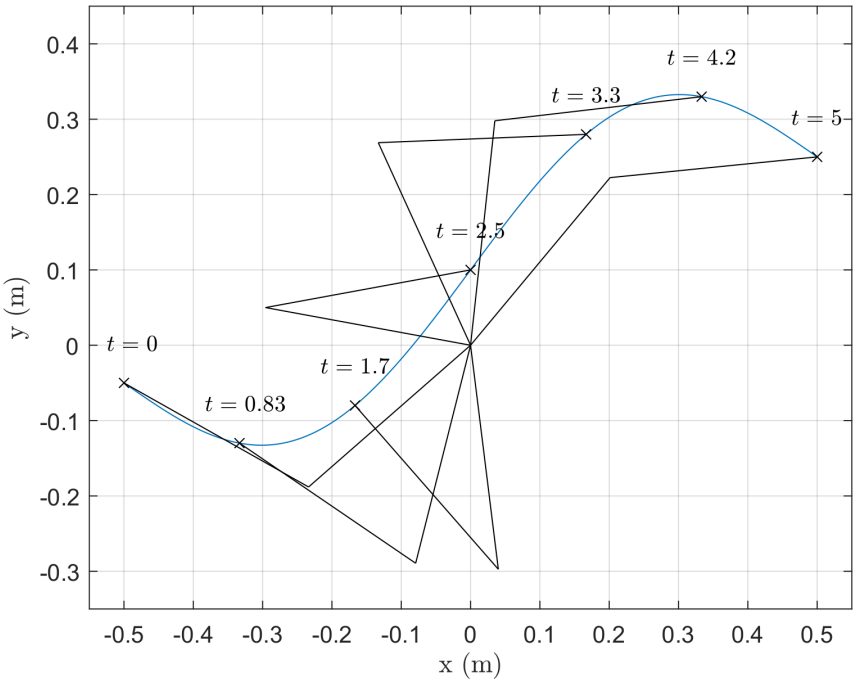


Figure 4.4: Tool Center Point Reference Trajectory [25]

The TCP trajectory shall be tracked despite the presence of disturbances. Hence a linear quadratic regulator is implemented for the nominal LTV system by minimizing the quadratic cost [cf. Equation (2.7)]

$$J(u) = x^T(T)Q_T x(T) + \int_0^T x^T(t)Qx(t) + u^T(t)Ru(t)\,\mathrm{d}t \tag{4.22}$$

with $x = \Theta - \Theta^*$, $u = \tau - \tau^*$, $Q = \mathrm{diag}(10, 1, 10, 1)$, $R = I$, and $Q_T = \mathrm{diag}(1, 0.1, 1, 0.1)$.

The optimal feedback controller is synthesized by solving the associated Riccati differential equation [cf. Equation (2.11)].

Now consider the manipulator's mass and inertia parameters (4.21) as uncertain. Take $\delta = [I_1, I_2, m_1, m_2, r_1, r_2]^T$ as a random vector, where each component is distributed uniformly around its mean $\overline{\delta}_i$, i.e., $\delta_i \sim \mathcal{U}(0.5\overline{\delta}_i, 1.5\overline{\delta}_i)$. The manipulator's mean parameters are $\overline{I}_1 = 0.09$ kg $\cdot$ m^2, $\overline{I}_2 = 0.06$ kg $\cdot$ m^2, $\overline{m}_1 = 3$ kg, $\overline{m}_2 = 2$ kg, $\overline{r}_1 = \overline{r}_2 = 0.15$ m. Both links have length $l_1 = l_2 = 0.3$ m. Within the present example, the analysis goal is to quantify the perturbed closed-loop trajectories due to the modeled parametric uncertainty and a constant torque disturbance added at the input u. We focus on mean and standard deviation of the random linear perturbations $x(t, \delta)$ around the target trajectory.

Three analysis methods are compared in the following:

- CGK: conventional Galerkin projection based PCE of the uncertain LTV system (Section 4.2).
- LFTGK: LFT Galerkin projection based PCE (Section 4.3) of the uncertain system in linear fractional representation.
- MC: Monte Carlo sampling-based estimation of the mean

$$\mathbb{E}[x(t,\cdot)] \approx \langle x(t,\cdot)\rangle_N := \frac{1}{N}\sum_{i=1}^{N} x(t,\delta^{(i)}) \tag{4.23}$$

  and standard deviation

$$\sqrt{\mathbb{V}[(x_j(t,\cdot)]} \approx \sigma[x_j(t,\cdot)] := \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left(x_j(t,\delta^{(i)}) - \langle x(t,\cdot)\rangle_N\right)^2}. \tag{4.24}$$

The subsequent computations are all conducted on a standard desktop PC with 4 GHz Intel i7 CPU and 16 GB RAM. The Julia toolbox *PolyChaos.jl* [51] is employed for computing the Galerkin projection. Time integrals are solved numerically using a standard Runge-Kutta scheme, e.g., `Tsit5`, implemented in [60] with relative tolerance 10^{-3} and absolute tolerance 10^{-6}.

Plots of mean and standard deviation fields as computed by LFTGK [cf. Equations (3.39) and (3.41)] are displayed in Figure 4.5. The PCE is truncated by maximum total polynomial degree truncation with a total degree $d = 3$. Thus the number of terms in the expansion is $L = 84$, see Equation (3.36). Applying CGK yields identical approximations of the mean and standard deviation (up to numerical accuracy).

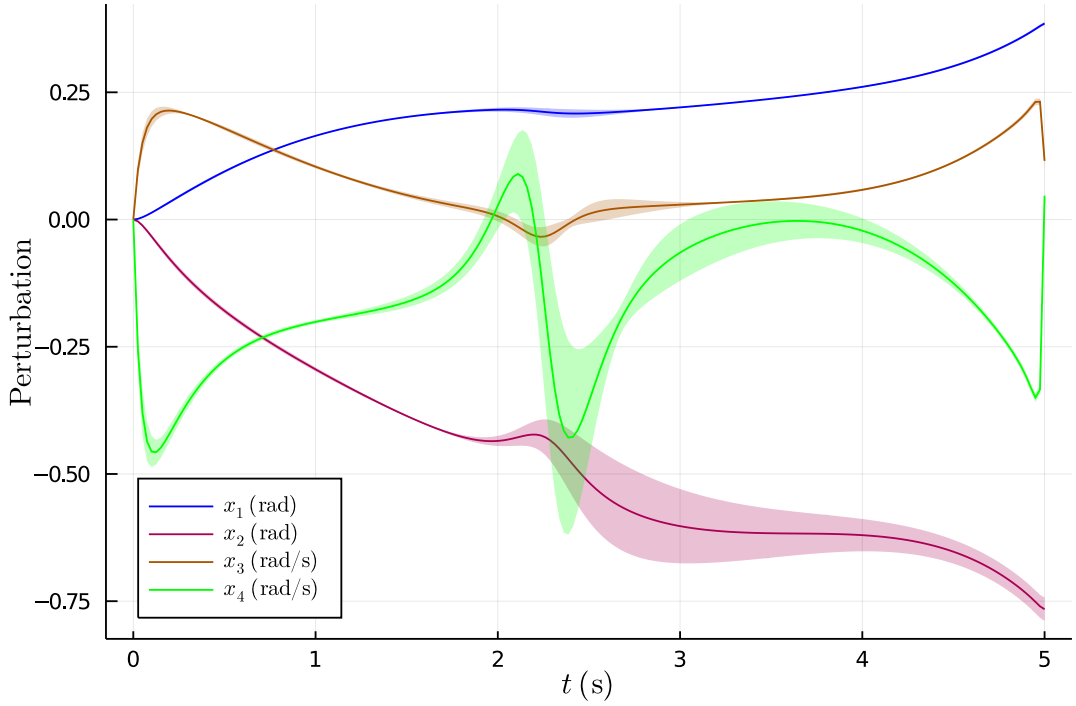


Figure 4.5: Mean and Standard Deviation Field of $d = 3$ LFTGK PCE [25]

However, in this example, it is necessary to compute multiplication tensors of order up to 5 for the conventional Galerkin projection. This can be seen by inserting the series expansion and projecting inertia terms such as $\langle m_1 r_1^2 \ddot{\theta}_1, \psi_\alpha \rangle$. The calculation of the required multiplication tensors takes 2 h. In practice, this is a reusable computation performed once before simulating the polynomial chaos expanded system. In contrast, the LFTGK solely requires computation of a third-order multiplication tensor. In particular, the Galerkin projection of the uncertain LFT system has a decoupled and sparse structure, as shown in Section 4.3. The computation time required for the especially sparse third-order multiplication tensor is therefore rather low (less than 1 s)

Table 4.1: Benchmark of CGK, LFTGK, and MC [25]

	Dimension	$\|e_m(t)\|_{\mathcal{L}_\infty}$	$\|e_\sigma(t)\|_{\mathcal{L}_\infty}$	Computation time
CGK	$d = 1$,	1.4×10^{-3}	1.7×10^{-2}	1 s
LFTGK	$L = 7$	2.3×10^{-3}	1.8×10^{-2}	1 s
MC	$N = 7$	5.3×10^{-2}	1.0×10^{-1}	2 s
CGK	$d = 2$,	1.5×10^{-3}	1.6×10^{-3}	11 s
LFTGK	$L = 28$	2.0×10^{-3}	2.6×10^{-3}	6 s
MC	$N = 50$	2.6×10^{-2}	1.5×10^{-2}	11 s
CGK	$d = 3$,	1.5×10^{-3}	4.9×10^{-4}	60 s
LFTGK	$L = 84$	1.9×10^{-3}	1.8×10^{-3}	16 s
MC	$N = 250$	2.5×10^{-3}	1.9×10^{-3}	63 s

compared to CGK.

The three analysis methods are benchmarked in Table 4.1. Different total polynomial degrees $d \in \{1, 2, 3\}$ are compared with a standard Monte Carlo estimator, whose sample set dimension N is scaled appropriately. Mean and standard deviation of a MC estimator with $N_{ref} = 10^4$ are used as an unbiased reference solution. The errors with respect to this MC reference are computed for the mean field [denoted $e_m(t)$] and standard deviation [denoted $e_\sigma(t)$]. The $\mathcal{L}_\infty$ signal norm, measuring the size of the error by $\|e(t)\|_{\mathcal{L}_\infty} = \max_{i=1,\dots,4} \sup_{t\in[0,T]} |e_i(t)|$, is displayed. The last column lists the computation time for numerical integration of the respective (system of) linear differential equations, excluding the calculation of inner products required by the Galerkin projection based methods. Notice that the $d = 3$ Galerkin projected system of equations require (less than) 1 min to solve, in contrast to 37 min for the $N_{ref} = 10^4$ MC samples. Comparing LFTGK to the fully coupled system of equations resulting from CGK, the decoupled sparse structure of LFTGK can be seen to yield computational advantages.

Finally, Figure 4.6 conducts a consistency check of the $d = 3$ LFTGK by comparing to three sampling-based methods: MC, Latin Hypercube Sampling (LHS), and Sobol' sequence. The maximum absolute difference over time in the approximation of $\langle x_4(t, \cdot)\rangle$ and $\sigma[x_4(t, \cdot)]$, cf. Equations (4.23) and (4.24), is plotted for increasing dimension N

of the sampling algorithms. The coordinate x_4 is selected since the highest variance and largest approximation errors were observed in this component. As expected, all sampling-based approaches converge roughly proportional to $N^{-1/2}$ (indicated by the red line) to the $d = 3$ LFTGK. This behavior gives confidence and trust in the LFTGK approach.

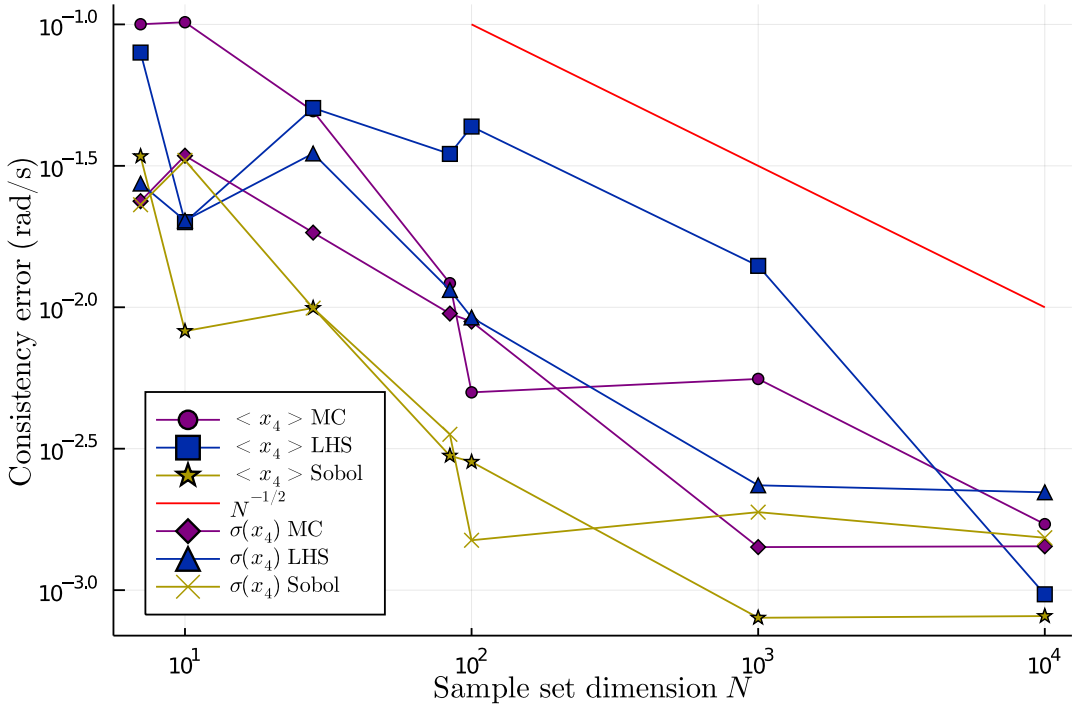


Figure 4.6: Consistency Check of $d = 3$ LFTGK [25]

In summary, separating the deterministic from the stochastic parts by a LFT before applying Galerkin projection can lead to computational savings and numerical advantages. Galerkin projection based PCEs (i.e., both CGK and LFTGK) are advantageous especially when only the deterministic problem part changes, e.g., a different reference trajectory or new controller is to be analyzed. In contrast to sampling-based, non-intrusive black-box methods, which have to rerun the complete sampling campaign, the PCEs based on CGK and LFTGK may provide prompt and accurate updated estimates. Further, the LFT lightens the workload of Galerkin projection in separating the deterministic from the uncertain parts of the problem.

4.5 Equivalence of Gaussian Quadrature and LFT Galerkin Projection

We now present the main theoretical results of this thesis. Large parts of this section are based on the author's article [28]. In order to simplify the proofs, we initially assume an orthonormal polynomial basis and a single uncertain parameter, i.e., $n_\delta = 1$. The results are extended to general uncertain linear systems in Section 4.5.2. Subsequently, the theoretical findings are shown to hold for an orthogonal polynomial basis in Section 4.5.3 and also for the multivariate case $n_\delta > 1$ in Section 4.5.4. The remaining Sections 4.5.5 and 4.5.6 illustrate the theory by means of two examples.

4.5.1 Main Theoretical Result

To set the stage, consider the uncertain linear time-invariant system

$$\dot{x}(t,\delta) = A(\delta)x(t,\delta). \tag{4.25}$$

Throughout this section, we assume an orthonormal polynomial expansion basis. The conventional intrusive Galerkin approach approximates the PCE coefficients

$$X(t) = \langle x(t,\cdot) \otimes \Psi \rangle \tag{4.26}$$

by directly projecting Equation (4.25) along the lines of Section 4.2:

$$\dot{X}(t) = \mathcal{A}X(t), \qquad \mathcal{A} := \langle A \otimes (\Psi\Psi^T) \rangle \in \mathbb{R}^{(n_x L)\times(n_x L)}. \tag{4.27}$$

In contrast, the non-intrusive method of Gaussian quadrature (see Section 3.5.1) determines the expansion coefficients by numerical integration:

$$X(t) = \langle x(t,\cdot) \otimes \Psi \rangle \approx \sum_{\alpha=1}^{N} w_\alpha \; x(t,\delta^{(\alpha)}) \otimes \Psi(\delta^{(\alpha)}). \tag{4.28}$$

Now assume the system (4.25) can be written as a linear fractional transformation

$$\begin{aligned}\begin{bmatrix}\dot{x}(t,\delta)\\ v(t,\delta)\end{bmatrix} &= \begin{bmatrix}M_{11} & M_{12}\\ M_{21} & M_{22}\end{bmatrix}\begin{bmatrix}x(t,\delta)\\ w(t,\delta)\end{bmatrix}\\ w(t,\delta) &= \Delta(\delta)v(t,\delta).\end{aligned} \tag{4.29}$$

The PCE given by Galerkin projection of the system's LFT is (see Section 4.3):

$$\begin{aligned}\begin{bmatrix}\dot{X}(t)\\ V(t)\end{bmatrix} &= \begin{bmatrix}M_{11}\otimes I_L & M_{12}\otimes I_L\\ M_{21}\otimes I_L & M_{22}\otimes I_L\end{bmatrix}\begin{bmatrix}X(t)\\ W(t)\end{bmatrix}\\ W(t) &= \underbrace{\langle \Delta\otimes(\Psi\Psi^T)\rangle}_{=:\Delta_\Pi} V(t).\end{aligned} \tag{4.30}$$

By closing the expanded LFT (4.30) — in contrast to Equation (4.27) — the expansion coefficients (4.26) are approximated by

$$\dot{X}(t) = \mathcal{A}_\Delta X(t), \qquad \mathcal{A}_\Delta := \mathcal{F}_l(M\otimes I_L, \Delta_\Pi). \tag{4.31}$$

Therein, the LFT Galerkin projected system matrix $\mathcal{A}_\Delta \in \mathbb{R}^{(n_x L)\times(n_x L)}$ is determined by the lower LFT

$$\mathcal{F}_l(M\otimes I_L,\Delta_\Pi) = M_{11}\otimes I_L + (M_{12}\otimes I_L)\underbrace{\Delta_\Pi[I-(M_{22}\otimes I_L)\Delta_\Pi]^{-1}}_{=:\partial\Delta_\Pi}(M_{21}\otimes I_L). \tag{4.32}$$

The perturbation term $\partial\Delta_\Pi$ is defined for later reference.

The main point of this section is to prove that the approximations (4.28) and (4.31) of $X(t)$ are equivalent. In order to build up to this main result, in the following, it is first shown that the eigensystem of $\mathcal{A}_\Delta$ is made up of realizations of $A(\delta)$ evaluated at the Gauss quadrature points. This is important, since generally speaking, conventional Galerkin projection may not necessarily yield meaningful realizations of the original system. There even exist mathematical examples of $A(\delta)$ stable $\forall\delta\in\mathcal{D}$ with an unstable conventional Galerkin projection (4.27). See Section 4.5.5 for details. This is alleviated completely by LFT Galerkin projection (4.31), which preserves the eigenvalues and eigenvectors of the original system. Hence, stability is preserved. The following theorem,

including the proof, is taken from the author's work in [28].

Theorem 4.1 (Gaussian Eigensystem Preservation [28])**.** *Let the lower LFT of* $A(\delta)$ *be given by*

$$A(\delta) = \mathcal{F}_l\{M, \Delta(\delta)\}.$$

For $\alpha = 1, ..., L$*, denote by* $\delta^{(\alpha)}$ *the* L *realizations of* $\delta \in \mathcal{D}$ *given by the distinct Gauss quadrature nodes of* δ*. For each* α*, let the* n_x *eigenvalues and eigenvectors of* $A(\delta^{(\alpha)})$ *be given by*

$$A(\delta^{(\alpha)})x_k^{(\alpha)} = \lambda_k^{(\alpha)}x_k^{(\alpha)}, \quad k = 1, ..., n_x.$$

Then the eigenvalues and eigenvectors of the LFT projected system matrix $\mathcal{A}_\Delta$ *defined in* (4.31) *are given by*

$$\mathcal{A}_\Delta\big[x_k^{(\alpha)} \otimes \Psi(\delta^{(\alpha)})\big] = \lambda_k^{(\alpha)}\big[x_k^{(\alpha)} \otimes \Psi(\delta^{(\alpha)})\big]$$

for $\alpha = 1, ..., L$ *and* $k = 1, ..., n_x$*.*

Proof [28]. In the interest of clarity, assume δ is a scalar, i.e., $n_\delta = 1$. The proof is generalized to orthogonal polynomials in Section 4.5.3 and then in Section 4.5.4 to the case $n_\delta > 1$. For $n_\delta = 1$, the Δ-matrix in the LFT $A(\delta) = \mathcal{F}_l\{M, \Delta(\delta)\}$ reduces to

$$\Delta(\delta) = \delta_1 I_{m_1} \tag{4.33}$$

with $\delta = \delta_1$ and multiplicity $m_1 = n_\Delta$. In this case, $\Delta_\Pi = \langle \Delta \otimes (\Psi\Psi^T)\rangle$ is given by the block-diagonal matrix

$$\Delta_\Pi = I_{n_\Delta} \otimes \langle \delta\Psi\Psi^T\rangle. \tag{4.34}$$

It is known, see [76] or Section 4.5.3, that the eigenvalues of $\langle \delta\Psi\Psi^T\rangle$ are distinct realizations $\delta^{(\alpha)}$ of δ corresponding to the Gauss quadrature nodes of order L. The corresponding eigenvectors are $\Psi(\delta^{(\alpha)})$, such that

$$\langle \delta\Psi\Psi^T\rangle\Psi(\delta^{(\alpha)}) = \delta^{(\alpha)}\Psi(\delta^{(\alpha)}). \tag{4.35}$$

Therefore, the eigenvalues of each block of Δ_Π are the distinct realizations $\delta^{(\alpha)}$ as well.

Hence, Δ_Π can be diagonalized by

$$T = I_{n_\Delta} \otimes V, \quad \Delta_\Pi = T(I_{n_\Delta} \otimes \Lambda)T^{-1} \tag{4.36}$$

where $V = [\Psi(\delta^{(\alpha)})]$ concatenates horizontally the column eigenvectors $\Psi(\delta^{(\alpha)})$ and $\Lambda = \text{diag}[\delta^{(\alpha)}]$ for $\alpha = 1, ..., L$. Using the diagonalization (4.36), note that the term $\partial\Delta_\Pi$ in the expanded LFT (4.32) can be "block-diagonalized", meaning

$$\begin{aligned}
\partial\Delta_\Pi &= \Delta_\Pi[I - (M_{22} \otimes I_L)\Delta_\Pi]^{-1} \\
&= T(I_{n_\Delta}\otimes\Lambda)T^{-1}[TT^{-1} - (M_{22}\otimes I_L)T(I_{n_\Delta}\otimes\Lambda)T^{-1}]^{-1} \\
&= T(I_{n_\Delta}\otimes\Lambda)[T - (M_{22}\otimes I_L)T(I_{n_\Delta}\otimes\Lambda)]^{-1} \\
&= T(I_{n_\Delta}\otimes\Lambda)[T - (M_{22}\otimes I_L)(I_{n_\Delta}\otimes V)(I_{n_\Delta}\otimes\Lambda)]^{-1} \\
&= T(I_{n_\Delta}\otimes\Lambda)[T - (I_{n_\Delta}\otimes V)(M_{22}\otimes I_L)(I_{n_\Delta}\otimes\Lambda)]^{-1} \\
&= T(I_{n_\Delta}\otimes\Lambda)[I - (M_{22}\otimes I_L)(I_{n_\Delta}\otimes\Lambda)]^{-1}T^{-1} \\
&= T(I_{n_\Delta}\otimes\Lambda)(I - M_{22}\otimes\Lambda)^{-1}T^{-1}.
\end{aligned} \tag{4.37}$$

Denote by $e_\alpha \in \mathbb{R}^L$ the L-dimensional unit vector with all zeros except the αth component. Define $E_\alpha := e_\alpha e_\alpha^T$. Then the term $(I - M_{22} \otimes \Lambda)^{-1}$ in Equation (4.37) can be written as

$$\begin{aligned}
(I - M_{22} \otimes \Lambda)^{-1} &= \left[\Big(\sum_{\alpha=1}^{L} I_{n_\Delta} \otimes E_\alpha \Big) - M_{22} \otimes \Big(\sum_{\alpha=1}^{L} \delta^{(\alpha)} E_\alpha \Big) \right]^{-1} \\
&= \left[\sum_{\alpha=1}^{L} \big(I_{n_\Delta} - M_{22}\delta^{(\alpha)} \big) \otimes E_\alpha \right]^{-1} \\
&= \sum_{\alpha=1}^{L} \big(I_{n_\Delta} - M_{22}\delta^{(\alpha)} \big)^{-1} \otimes E_\alpha.
\end{aligned} \tag{4.38}$$

The last equality can be checked by multiplying

$$\left[\sum_{\alpha=1}^{L} \big(I_{n_\Delta} - M_{22}\delta^{(\alpha)} \big)^{-1} \otimes E_\alpha \right] \left[\sum_{\beta=1}^{L} \big(I_{n_\Delta} - M_{22}\delta^{(\beta)} \big) \otimes E_\beta \right] \tag{4.39}$$

which yields the identity matrix. Now define by $T_\alpha : \mathbb{R}^{n_x} \mapsto \mathbb{R}^{n_x L}$ the linear submap

of $\mathcal{A}_\Delta$

$$\begin{aligned} T_\alpha(x) &:= \mathcal{A}_\Delta[x \otimes \Psi(\delta^{(\alpha)})] = \mathcal{F}_l(M \otimes I_L, \Delta_\Pi)[x \otimes \Psi(\delta^{(\alpha)})] \\ &= [M_{11} \otimes I_L + (M_{12} \otimes I_L)\partial\Delta_\Pi(M_{21} \otimes I_L)][x \otimes \Psi(\delta^{(\alpha)})] \\ &= (M_{11}x) \otimes \Psi(\delta^{(\alpha)}) + (M_{12} \otimes I_L)\partial\Delta_\Pi[M_{21}x \otimes \Psi(\delta^{(\alpha)})] \\ &= (M_{11}x) \otimes \Psi(\delta^{(\alpha)}) + (M_{12} \otimes I_L)\partial\Delta_\Pi[I_{n_\Delta} \otimes \Psi(\delta^{(\alpha)})](M_{21}x \otimes 1). \end{aligned} \tag{4.40}$$

Therein, using Equation (4.37) and $T = I_{n_\Delta} \otimes V$

$$\begin{aligned} \partial\Delta_\Pi[I_{n_\Delta} \otimes \Psi(\delta^{(\alpha)})] &= (I_{n_\Delta} \otimes V\Lambda)(I - M_{22} \otimes \Lambda)^{-1}(I_{n_\Delta} \otimes V^{-1})[I_{n_\Delta} \otimes \Psi(\delta^{(\alpha)})] \\ &= (I_{n_\Delta} \otimes V\Lambda)(I - M_{22} \otimes \Lambda)^{-1}(I_{n_\Delta} \otimes e_\alpha). \end{aligned} \tag{4.41}$$

Inserting Equation (4.38) yields

$$\begin{aligned} \partial\Delta_\Pi[I_{n_\Delta} \otimes \Psi(\delta^{(\alpha)})] &= (I_{n_\Delta} \otimes V\Lambda)\left[\sum_{\alpha=1}^{L} \left(I_{n_\Delta} - M_{22}\delta^{(\alpha)}\right)^{-1} \otimes E_\alpha\right](I_{n_\Delta} \otimes e_\alpha) \\ &= (I_{n_\Delta} \otimes V\Lambda)\left[(I - M_{22}\delta^{(\alpha)})^{-1} \otimes e_\alpha\right] \\ &= \delta^{(\alpha)}(I - M_{22}\delta^{(\alpha)})^{-1} \otimes \Psi(\delta^{(\alpha)}). \end{aligned} \tag{4.42}$$

Plugging (4.42) into (4.40), $T_\alpha(x)$ becomes

$$\begin{aligned} T_\alpha(x) &= (M_{11}x) \otimes \Psi(\delta^{(\alpha)}) + (M_{12} \otimes I_L)\left[\delta^{(\alpha)}(I - M_{22}\delta^{(\alpha)})^{-1} \otimes \Psi(\delta^{(\alpha)})\right](M_{21}x \otimes 1) \\ &= \left[M_{11}x + M_{12}\delta^{(\alpha)}(I - M_{22}\delta^{(\alpha)})^{-1}M_{21}x\right] \otimes \Psi(\delta^{(\alpha)}). \end{aligned} \tag{4.43}$$

In (4.43), for a specific α, i.e., eigenvector $\Psi(\delta^{(\alpha)})$, one picks out a realization of the LFT's perturbation term

$$M_{12}\Delta(\delta^{(\alpha)})[I - M_{22}\Delta(\delta^{(\alpha)})]^{-1}M_{21}. \tag{4.44}$$

Adding the nominal map M_{11} then yields a realization $A(\delta^{(\alpha)})$. Denote by $\lambda_k^{(\alpha)}$ and $x_k^{(\alpha)}$ the eigenvalues and eigenvectors of $A(\delta^{(\alpha)})$

$$A(\delta^{(\alpha)})x_k^{(\alpha)} = \lambda_k^{(\alpha)}x_k^{(\alpha)} \tag{4.45}$$

for $k = 1, ..., n_x$. Clearly, by definition of T_α:

$$\mathcal{A}_\Delta[x_k^{(\alpha)} \otimes \Psi(\delta^{(\alpha)})] = T_\alpha(x_k^{(\alpha)}) = \lambda_k^{(\alpha)}[x_k^{(\alpha)} \otimes \Psi(\delta^{(\alpha)})]. \tag{4.46}$$

This yields the desired statement of the theorem. □

Consequently, the eigenvalues and eigenvectors of $A(\delta^{(\alpha)})$ are preserved within the LFT Galerkin projection $\mathcal{A}_\Delta$. The subsequent Corollary follows immediately. It is briefly stated in order to simplify the proof of our main result afterwards.

Corollary 4.1. $\mathcal{A}_\Delta[I_{n_x} \otimes \Psi(\delta^{(\alpha)})] = A(\delta^{(\alpha)}) \otimes \Psi(\delta^{(\alpha)})$.

Proof. The proof is identical to the derivation of Equations (4.40)–(4.43). The vector x merely needs to be replaced by the identity matrix I_{n_x}. □

Our main theoretical result is presented in the following theorem, taken from [28].

Theorem 4.2. *The expansion coefficients $X(t)$ determined by Gaussian quadrature* (4.28) *are equivalent to $X(t)$ obtained by LFT Galerkin projection* (4.31).

Proof [28]. Denote by $X(0)$ the expansion coefficients of the initial condition $x(0, \delta)$ to the uncertain system (4.25)

$$X(0) = \langle x(0, \cdot) \otimes \Psi \rangle. \tag{4.47}$$

Without loss of generality, assume the projection (4.47) is calculated by Gaussian quadrature

$$X(0) = \sum_{\alpha=1}^{L} w_\alpha \; x(0, \delta^{(\alpha)}) \otimes \Psi(\delta^{(\alpha)}). \tag{4.48}$$

This assumption is not a restriction. In essence, the initial condition $X(0)$ for both approaches (4.28) and (4.31) simply needs to be the same. Clearly, since (4.31) is a linear system, the expansion coefficients determined by LFT Galerkin projection obey

$$X(t) = e^{\mathcal{A}_\Delta t} X(0). \tag{4.49}$$

Using Corollary 4.1, note that

$$\begin{aligned} e^{\mathcal{A}_\Delta t}[I_{n_x} \otimes \Psi(\delta^{(\alpha)})] &= \sum_{k=0}^{\infty} \frac{(\mathcal{A}_\Delta t)^k}{k!}[I_{n_x} \otimes \Psi(\delta^{(\alpha)})] \\ &= \Big[I + \mathcal{A}_\Delta t + \frac{(\mathcal{A}_\Delta t)^2}{2} + \dots\Big][I_{n_x} \otimes \Psi(\delta^{(\alpha)})] \\ &= I_{n_x} \otimes \Psi(\delta^{(\alpha)}) + A(\delta^{(\alpha)})t \otimes \Psi(\delta^{(\alpha)}) + \frac{[A(\delta^{(\alpha)})t]^2}{2} \otimes \Psi(\delta^{(\alpha)}) + \dots \\ &= e^{A(\delta^{(\alpha)})t} \otimes \Psi(\delta^{(\alpha)}). \end{aligned} \tag{4.50}$$

Plugging Equation (4.48) into (4.49) yields

$$X(t) = e^{\mathcal{A}_\Delta t} \sum_{\alpha=1}^{L} w_\alpha \; x(0, \delta^{(\alpha)}) \otimes \Psi(\delta^{(\alpha)}) = \sum_{\alpha=1}^{L} w_\alpha \; e^{\mathcal{A}_\Delta t}[I_{n_x} \otimes \Psi(\delta^{(\alpha)})]x(0, \delta^{(\alpha)}). \tag{4.51}$$

Inserting Equation (4.50), we obtain

$$\begin{aligned} X(t) &= \sum_{\alpha=1}^{L} w_\alpha \; \big[e^{A(\delta^{(\alpha)})t} \otimes \Psi(\delta^{(\alpha)})\big] x(0, \delta^{(\alpha)}) \\ &= \sum_{\alpha=1}^{L} w_\alpha \; \big[e^{A(\delta^{(\alpha)})t} x(0, \delta^{(\alpha)})\big] \otimes \Psi(\delta^{(\alpha)}) \\ &= \sum_{\alpha=1}^{L} w_\alpha \; x(t, \delta^{(\alpha)}) \otimes \Psi(\delta^{(\alpha)}) \end{aligned} \tag{4.52}$$

which is clearly equivalent to Gaussian quadrature (4.28) for $N = L$. □

In conclusion, for (systems of) uncertain linear differential equations with rational parameter-dependence, applying LFT Galerkin projection is equivalent to applying Gaussian quadrature.

4.5.2 General Uncertain Linear Systems

Within this section, we show that the previous results generalize to uncertain linear systems with an input u. For this task, it does not matter if the system matrices are time-varying or not. Thus, to simplify the notation, we assume the time-invariant case.

Consider an uncertain LTI system (see Section 2.1.2):

$$\begin{aligned} \dot{x} &= A(\delta)x + B(\delta)u \\ y &= C(\delta)x + D(\delta)u. \end{aligned} \tag{4.53}$$

According to Section 4.3, assume Equation (4.53) can be written as a linear fractional transformation

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ v \\ y \end{bmatrix} &= \begin{bmatrix} A_M & B_w & B_u \\ C_v & D_{vw} & D_{vu} \\ C_y & D_{yw} & D_{yu} \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix} \\ w &= \Delta v \end{aligned} \tag{4.54}$$

with LFT Galerkin projection

$$\begin{aligned} \begin{bmatrix} \dot{X} \\ V \\ Y \end{bmatrix} &= \begin{bmatrix} A_M \otimes I_L & B_w \otimes I_L & B_u \otimes e_1 \\ C_v \otimes I_L & D_{vw} \otimes I_L & D_{vu} \otimes e_1 \\ C_y \otimes I_L & D_{yw} \otimes I_L & D_{yu} \otimes e_1 \end{bmatrix} \begin{bmatrix} X \\ W \\ u \end{bmatrix} \\ W &= \Delta_\Pi V. \end{aligned} \tag{4.55}$$

Define the following matrices by closing the LFT (4.55):

$$\begin{aligned} \mathcal{A}_\Delta &:= \mathcal{F}_l\left(\begin{bmatrix} A_M & B_w \\ C_v & D_{vw} \end{bmatrix} \otimes I_L,\ \Delta_\Pi \right) \\ &= A_M \otimes I_L + (B_w \otimes I_L)\Delta_\Pi[I - (D_{vw} \otimes I_L)\Delta_\Pi]^{-1}(C_v \otimes I_L) \end{aligned} \tag{4.56}$$

$$\begin{aligned} \mathcal{B}_\Delta &:= \mathcal{F}_l\left(\begin{bmatrix} B_u \otimes e_1 & B_w \otimes I_L \\ D_{vu} \otimes e_1 & D_{vw} \otimes I_L \end{bmatrix},\ \Delta_\Pi \right) \\ &= B_u \otimes e_1 + (B_w \otimes I_L)\Delta_\Pi[I - (D_{vw} \otimes I_L)\Delta_\Pi]^{-1}(D_{vu} \otimes e_1) \end{aligned} \tag{4.57}$$

$$\begin{aligned} \mathcal{C}_\Delta &:= \mathcal{F}_l\left(\begin{bmatrix} C_y & D_{yw} \\ C_v & D_{vw} \end{bmatrix} \otimes I_L,\ \Delta_\Pi \right) \\ &= C_y \otimes I_L + (D_{yw} \otimes I_L)\Delta_\Pi[I - (D_{vw} \otimes I_L)\Delta_\Pi]^{-1}(C_v \otimes I_L) \end{aligned} \tag{4.58}$$

$$\begin{aligned}\mathcal{D}_\Delta &:= \mathcal{F}_l\left(\begin{bmatrix} D_{yu}\otimes e_1 & D_{yw}\otimes I_L \\ D_{vu}\otimes e_1 & D_{vw}\otimes I_L\end{bmatrix},\ \Delta_\Pi\right) \\ &= D_{yu}\otimes e_1 + (D_{yw}\otimes I_L)\Delta_\Pi[I-(D_{vw}\otimes I_L)\Delta_\Pi]^{-1}(D_{vu}\otimes e_1).\end{aligned} \tag{4.59}$$

The stochastic modes of the closed LFT Galerkin projection are thus given by

$$\begin{bmatrix}\dot{X}\\ Y\end{bmatrix} = \begin{bmatrix}\mathcal{A}_\Delta & \mathcal{B}_\Delta \\ \mathcal{C}_\Delta & \mathcal{D}_\Delta\end{bmatrix}\begin{bmatrix}X\\ u\end{bmatrix}. \tag{4.60}$$

Similar to Corollary 4.1, it is straightforward to show that LFT projected linear systems (4.60) can be "block-diagonalized", as stated in the following corollary.

Corollary 4.2.

$$\begin{bmatrix}\mathcal{A}_\Delta & \mathcal{B}_\Delta \\ \mathcal{C}_\Delta & \mathcal{D}_\Delta\end{bmatrix}\left(\begin{bmatrix}I_{n_x} & 0\\ 0 & I_{n_u}\end{bmatrix}\otimes\Psi(\delta^{(\alpha)})\right) = \begin{bmatrix}A(\delta^{(\alpha)}) & B(\delta^{(\alpha)}) \\ C(\delta^{(\alpha)}) & D(\delta^{(\alpha)})\end{bmatrix}\otimes\Psi(\delta^{(\alpha)}).$$

Proof. The proof is analogous to the derivation of Equations (4.40)–(4.43) in the proof of Theorem 4.1. It is therefore omitted. □

Using Corollary 4.2, it is a simple exercise to extend the proof of Theorem 4.2 to general uncertain LTI systems, i.e, to show that the stochastic modes obtained by LFT Galerkin projection (4.60) and by Gaussian quadrature are equivalent. In an even broader context, the results generalize to any linear system of differential equations with rational parameter-dependence. For example, this also applies to Lyapunov differential matrix equations. These appear, for instance, when computing the finite horizon $\mathcal{H}_2$-norm [19, 33]. However, it does not apply to Riccati differential equations as considered in the subsequent chapter due to the quadratic nonlinearity.

4.5.3 Extension to Orthogonal Polynomials

Previously, Theorem 4.1 is proven assuming an orthonormal polynomial basis and $n_\delta = 1$. The proof is generalized here to orthogonal polynomials (similar to [28]) and to $n_\delta > 1$ in the subsequent section. It is sufficient to generalize solely the proof of Theorem 4.1, since the remaining results follow immediately as soon as Theorem 4.1 holds.

Assume the polynomial basis is orthogonal instead of orthonormal. The uncertainty δ is still considered as a scalar throughout this section, i.e., $n_\delta = 1$. Thus, Δ_Π in the expanded LFT (4.30) is then given by (see Section 4.3):

$$\Delta_\Pi = (I_{n_\Delta} \otimes \langle \Psi\Psi^T \rangle)^{-1} \langle \Delta \otimes (\Psi\Psi^T) \rangle = (I_{n_\Delta} \otimes \langle \Psi\Psi^T \rangle)^{-1} \langle \delta I_{n_\Delta} \otimes (\Psi\Psi^T) \rangle. \quad (4.61)$$

Note that this Δ_Π is not symmetric in contrast to the orthonormal case (Section 4.5.1). However, it can be shown, analogously to [76], that the $n_\Delta \cdot L$ eigenvalues of Δ_Π are the L distinct Gauss quadrature points of δ, repeated n_Δ times.

To see this, we need the insights of orthogonal polynomial theory presented in Sections 3.2.2 and 3.2.3. Rewrite the recurrence relations in Equation (3.11) as

$$\begin{aligned} \delta\psi_\alpha(\delta) &= \psi_{\alpha+1}(\delta) + a_\alpha \psi_\alpha(\delta) + b_\alpha \psi_{\alpha-1}(\delta) \\ \psi_0(\delta) &= 1 \\ \psi_{-1}(\delta) &= 0. \end{aligned} \quad (4.62)$$

Using the recursion (4.62), note that the term $\langle \delta\psi_\alpha\psi_\beta \rangle$ within (4.61) may be written as

$$\langle \delta\psi_\alpha\psi_\beta \rangle = \langle (\psi_{\alpha+1} + a_\alpha\psi_\alpha + b_\alpha\psi_{\alpha-1})\psi_\beta \rangle = \langle \psi_\beta^2 \rangle (b_\alpha \delta_{(\alpha-1)\beta} + a_\alpha \delta_{\alpha\beta} + \delta_{(\alpha+1)\beta}) \quad (4.63)$$

where $\delta_{\alpha\beta}$ denotes the Kronecker delta. Note further that $\langle \delta\Psi\Psi^T \rangle$ is symmetric since $\langle \delta\psi_\alpha\psi_\beta \rangle = \langle \delta\psi_\beta\psi_\alpha \rangle$. It can then be verified using Equation (4.63) that each block of Δ_Π is given by the *Jacobi matrix* $\mathcal{J}_L$ introduced in Section 3.2.3:

$$\langle \Psi\Psi^T \rangle^{-1} \langle \delta\Psi\Psi^T \rangle = \mathcal{J}_L = \begin{bmatrix} a_0 & 1 & & & \\ b_1 & a_1 & 1 & & \\ & b_2 & \ddots & \ddots & \\ & & \ddots & a_{L-2} & 1 \\ & & & b_{L-1} & a_{L-1} \end{bmatrix}. \quad (4.64)$$

Furthermore, it is shown in Section 3.2.3 that the eigenvalues of $\mathcal{J}_L$ are the L roots of the orthogonal polynomial $\psi_L(\delta)$. These roots are exactly the Gauss quadrature nodes of order L. It is also shown in Lemma 3.1 that the roots of $\psi_L(\delta)$ are all real and distinct.

Hence, the eigenvalues of $\mathcal{J}_L$ are real and distinct. It follows that the spectrum of $\mathcal{J}_L$ is simple, implying the eigenspace belonging to each eigenvalue is one-dimensional. Therefore, Δ_Π can be diagonalized just as in Equation (4.36). The remaining proof of Theorem 4.1 given in Section 4.5.1 holds unaltered.

4.5.4 Generalization to $n_\delta > 1$

Within this section, we extend the proof of Theorem 4.1 to the general case $n_\delta > 1$, based on [28]. In addition to assuming an orthogonal polynomial basis, we now consider δ as a random vector comprising n_δ independent random variables. According to Section 3.3.2, an appropriate multivariate orthogonal polynomial basis can then be built simply by taking products of the individual univariate orthogonal polynomials. As a concrete example, assume for $i = 1, ..., n_\delta$ each δ_i has an orthogonal (univariate) polynomial basis truncated by maximum polynomial degree d. Denote the vector concatenating the respective basis polynomials by

$$\Psi_i(\delta_i) = \left[\psi_0^{(i)}(\delta_i), \ldots, \psi_d^{(i)}(\delta_i)\right]^T \in \mathbb{R}^{d+1}. \tag{4.65}$$

The associated multivariate orthogonal polynomial basis is then given by

$$\Psi = \Psi_1 \otimes \ldots \otimes \Psi_{n_\delta} =: \bigotimes_{i=1}^{n_\delta} \Psi_i. \tag{4.66}$$

Clearly, $\Psi \in \mathbb{R}^L$ is of dimension $L = (d+1)^{n_\delta}$. Recall that the matrix

$$\Delta_\Pi = (I_{n_\Delta} \otimes \langle \Psi\Psi^T \rangle)^{-1} \langle \Delta \otimes (\Psi\Psi^T) \rangle \tag{4.67}$$

is made up of $i = 1, ..., n_\delta$ blocks

$$\langle \Psi\Psi^T \rangle^{-1} \langle \delta_i \Psi\Psi^T \rangle \tag{4.68}$$

possibly repeated with multiplicity m_i. Note that in Equation (4.68), we can integrate

$$\langle \delta_i \Psi\Psi^T \rangle = \langle \delta_i (\Psi_1\Psi_1^T) \otimes \ldots \otimes (\Psi_{n_\delta}\Psi_{n_\delta}^T) \rangle = \langle \delta_i \bigotimes_{j=1}^{n_\delta} (\Psi_j\Psi_j^T) \rangle \tag{4.69}$$

independently over each δ_j:

$$\langle \delta_i \Psi \Psi^T \rangle = \left[\bigotimes_{j=1}^{i-1} \langle \Psi_j \Psi_j^T \rangle \right] \otimes \langle \delta_i \Psi_i \Psi_i^T \rangle \otimes \left[\bigotimes_{j=i+1}^{n_\delta} \langle \Psi_j \Psi_j^T \rangle \right]. \tag{4.70}$$

Consistent with the previous section, denote by $\mathcal{J}_{d+1}^{(i)}$ the Jacobi matrix associated with the orthogonal polynomial basis Ψ_i for δ_i. It can be concluded that

$$\begin{aligned} \langle \Psi \Psi^T \rangle^{-1} \langle \delta_i \Psi \Psi^T \rangle &= \left[\bigotimes_{j=1}^{i-1} I_{d+1} \right] \otimes \mathcal{J}_{d+1}^{(i)} \otimes \left[\bigotimes_{j=i+1}^{n_\delta} I_{d+1} \right] \\ &= I_{d+1} \otimes \ldots \otimes I_{d+1} \otimes \mathcal{J}_{d+1}^{(i)} \otimes I_{d+1} \otimes \ldots \otimes I_{d+1}. \end{aligned} \tag{4.71}$$

The distinct eigenvalues of $\mathcal{J}_{d+1}^{(i)}$ are given by the univariate $(d+1)$-point Gauss quadrature rule with respect to the probability density of δ_i. Hence

$$\mathcal{J}_{d+1}^{(i)} = T_i \Lambda_i T_i^{-1} \tag{4.72}$$

can be diagonalized, where $T_i = [\Psi_i(\delta_i^{(\alpha_i)})]$ concatenates horizontally the column eigenvectors $\Psi_i(\delta_i^{(\alpha_i)})$ and $\Lambda_i = \text{diag}[\delta_i^{(\alpha_i)}]$ for $\alpha_i = 1, ..., d+1$. In conclusion, Δ_Π can be diagonalized as

$$\Delta_\Pi = T \Lambda T^{-1} \tag{4.73}$$

with

$$\begin{aligned} T &= \begin{bmatrix} I_{m_1} \otimes T_1 \otimes I_{d+1} \otimes \ldots \otimes I_{d+1} & & 0 \\ & \ddots & \\ 0 & & I_{m_{n_\delta}} \otimes I_{d+1} \otimes \ldots \otimes I_{d+1} \otimes T_{n_\delta} \end{bmatrix} \\ \Lambda &= \begin{bmatrix} I_{m_1} \otimes \Lambda_1 \otimes I_{d+1} \otimes \ldots \otimes I_{d+1} & & 0 \\ & \ddots & \\ 0 & & I_{m_{n_\delta}} \otimes I_{d+1} \otimes \ldots \otimes I_{d+1} \otimes \Lambda_{n_\delta} \end{bmatrix}. \end{aligned} \tag{4.74}$$

The remaining proof of Theorem 4.1 is analogous to the one given in Section 4.5.1. Consistently, $\partial\Delta_\Pi$ can again be "block-diagonalized", i.e.,

$$\begin{aligned}\partial\Delta_\Pi &= \Delta_\Pi[I-(M_{22}\otimes I_L)\Delta_\Pi]^{-1} = T\Lambda T^{-1}[TT^{-1}-(M_{22}\otimes I_L)T\Lambda T^{-1}]^{-1}\\ &= T\Lambda[T-(M_{22}\otimes I_L)T\Lambda]^{-1}\\ &= T\Lambda[I-(M_{22}\otimes I_L)\Lambda]^{-1}T^{-1}.\end{aligned} \tag{4.75}$$

Using a standard multi-index notation $\alpha = \alpha_1 \times \ldots \times \alpha_{n_\delta} \in \mathbb{N}_0^{n_\delta}$ (see e.g. [77]), define by $E_\alpha := E_{\alpha_1}\otimes\ldots\otimes E_{\alpha_{n_\delta}}$. Here, analogous to Section 4.5.1, denote $E_{\alpha_i} := e_{\alpha_i}e_{\alpha_i}^T$ where $e_{\alpha_i}\in\mathbb{R}^{d+1}$ is the unit vector with all zeros except the α_ith component. In the following equation, for notational simplicity, sums are indexed by a single scalar α. The relationship between multi-indices $\alpha\in\mathbb{N}_0^{n_\delta}$ and a single scalar index $\alpha\in\mathbb{N}_0$ is one to one assuming the polynomial basis is ordered, e.g., lexicographically. See e.g. [77] for details. The middle term in Equation (4.75) may then be rewritten as

$$\begin{aligned}&[I-(M_{22}\otimes I_L)\Lambda]^{-1} = \left[\sum_{\alpha=1}^{L} I_{n_\Delta}\otimes E_\alpha - (M_{22}\otimes I_L)\left(\sum_{\alpha=1}^{L}\Delta(\delta^{(\alpha)})\otimes E_\alpha\right)\right]^{-1}\\ &= \left[\sum_{\alpha=1}^{L} I_{n_\Delta}\otimes E_\alpha - \left(M_{22}\Delta(\delta^{(\alpha)})\right)\otimes E_\alpha\right]^{-1} = \left[\sum_{\alpha=1}^{L}\left(I_{n_\Delta}-M_{22}\Delta(\delta^{(\alpha)})\right)\otimes E_\alpha\right]^{-1}\\ &= \sum_{\alpha=1}^{L}\left[I_{n_\Delta}-M_{22}\Delta(\delta^{(\alpha)})\right]^{-1}\otimes E_\alpha.\end{aligned} \tag{4.76}$$

At this point, just as in Section 4.5.1, define by $T_\alpha:\mathbb{R}^{n_x}\mapsto\mathbb{R}^{n_xL}$ the linear submap of $\mathcal{A}_\Delta$

$$\begin{aligned}T_\alpha(x) &:= \mathcal{A}_\Delta[x\otimes\Psi(\delta^{(\alpha)})] = \mathcal{F}_l(M\otimes I_L,\Delta_\Pi)[x\otimes\Psi(\delta^{(\alpha)})]\\ &= (M_{11}x)\otimes\Psi(\delta^{(\alpha)}) + (M_{12}\otimes I_L)\partial\Delta_\Pi[I_{n_\Delta}\otimes\Psi(\delta^{(\alpha)})]M_{21}x.\end{aligned} \tag{4.77}$$

Herein, together with Equations (4.75) and (4.76), it can be shown that

$$\partial\Delta_\Pi[I_{n_\Delta}\otimes\Psi(\delta^{(\alpha)})] = \left\{\Delta(\delta^{(\alpha)})[I-M_{22}\Delta(\delta^{(\alpha)})]^{-1}\right\}\otimes\Psi(\delta^{(\alpha)}). \tag{4.78}$$

The rest of the proof is precisely as in Section 4.5.1. As a supplement, note that the validity of Theorem 4.1 remains untouched despite using different truncation schemes. For example, truncating by maximum total polynomial degree translates to simply deleting some rows and columns from the matrix Δ_Π considered throughout this section.

4.5.5 Illustrative Example

Within this section, we visualize our main theoretical results by means of an example given in [28]. Consider the system matrix

$$A(\delta) = \frac{1}{100}\begin{bmatrix} 128\delta^2-72\delta-32 & 295\delta^2-199\delta+4 & 165\delta^2-234\delta+46 \\ -82\delta^2-59\delta+270 & -266\delta^2+144\delta-73 & -147\delta^2-210\delta+286 \\ 70\delta^2+296\delta-80 & 43\delta^2+96\delta+8 & 15\delta^2+146\delta-251 \end{bmatrix} \tag{4.79}$$

originating from [59]. The eigenvalues of $A(\delta)$ ranging within $\delta \in [-1, 1]$ are indicated in Figure 4.7 by the colored lines. Eigenvalues with real part less than -1.3 are not displayed for brevity. The matrix (4.79) is asymptotically stable for all δ in the aforementioned parameter range. Therein, assume δ is distributed uniformly, such that the corresponding orthogonal polynomial basis is given by the Legendre polynomials. Note that for a polynomial expansion degree $d = 0$, the conventional Galerkin projection already generates an unstable system, see the top plot in Figure 4.7. The $d = 0$ projection (i.e., $L = 1$) is given simply by the expectation $\mathbb{E}[A]$. Clearly, this conventionally projected system does not yield a meaningful realization regarding the eigenvalues of the original system. It can also be observed for higher-order expansion degrees that the eigenvalues of the conventional Galerkin projection are not necessarily realizations of the actual eigenvalues. In this example, the conventionally Galerkin projected system remains unstable regardless of the expansion order.

In contrast, the eigenvalues of the LFT Galerkin projection are shown in the bottom plot of Figure 4.7 for different polynomial expansion degrees d. Here, the eigenvalues correspond to meaningful realizations of the actual system. It can be verified numerically that these are precisely the realizations given by the respective Gauss quadrature points of δ.

Figure 4.8 compares the two Galerkin projected systems in the time domain for increasing polynomial expansion degrees d. Visible are the approximations of $\mathbb{E}[x_1(t,\cdot)]$ given by the $\alpha = 0$th expansion coefficient. Both projected systems are always simulated starting from the same deterministic initial condition $x(0) = [-0.75, -0.65, 0.08]^T$. Evidently, all the conventional approximations diverge after some time, since the conventional Galerkin projection is unstable. The bottom plot shows the approximations of $\mathbb{E}[x_1(t,\cdot)]$ given by LFT Galerkin projection and Gaussian quadrature. Since these approximations are equivalent, the plotted lines lie above of each other. The maximum absolute difference between the approximations of $\mathbb{E}[x_1(t,\cdot)]$ yielded by LFT Galerkin projection and Gaussian quadrature is less than 10^{-14} for all t. This illustrates the statement of Theorem 4.2.

4.5.6 LTI Robust Performance Lower Bound

Within this section, we present an application to the robust performance analysis of uncertain LTI systems, taken from [28]. According to Section 2.3, robust performance can be certified by solving a parameter-dependent algebraic Riccati equation (ARE). This applies to several system norms appearing throughout control theory, e.g., the $\mathcal{H}_\infty$-norm. Equivalently, the associated Hamiltonian matrix $H(\delta)$ can be checked for eigenvalues on the imaginary axis, see Theorem 2.4. System norms are then often computed by finding the minimal scalar performance index γ while $H(\delta)$ does not have imaginary axis eigenvalues for all δ, since this is a requirement for the existence of a stabilizing solution to the ARE.

Since the eigenvalues of the LFT projected matrix are actual realizations of the original Hamiltonian, we have derived a lower bound for robust performance computed by solving algebraic Riccati equations. Assume an uncertain LTI system is given, depending on a parameter vector δ. If we project the associated random parameter-dependent Hamiltonian $H(\delta)$ according to the LFT Galerkin projection scheme, we end up with Gauss quadrature realizations of the actual eigenvalues. Hence, the projected Hamiltonian will have eigenvalues on the imaginary axis only if the actual Hamiltonian does so too. Thus, minimizing over γ yields a lower bound for the exact robust performance index.

The following example briefly illustrates the main gist of this concept. Consider an uncertain LTI system $G(\delta)$

$$\begin{aligned} \dot{x} &= A(\delta)x + Bu \\ y &= Cx \end{aligned} \tag{4.80}$$

where $A(\delta)$ is given in Equation (4.79) and $B = C = I_3$. For some $\gamma > 0$, the Hamiltonian matrix associated with the $\mathcal{H}_\infty$-norm $\|G\|_\infty$ is (see Section 2.3):

$$H(\delta) = \begin{bmatrix} A(\delta) & \gamma^{-2}BB^T \\ -C^TC & -A^T(\delta) \end{bmatrix}. \tag{4.81}$$

Figure 4.9 shows different approximations of the robust $\mathcal{H}_\infty$-norm

$$\max_{\delta\in[-1,1]} \|G(\delta)\|_\infty = \min \gamma \quad \text{s.t. } \det[j\omega I - H(\delta)] \neq 0 \quad \forall(\delta,\omega) \in [-1,1] \times \mathbb{R}. \tag{4.82}$$

As a comparison, mean and standard deviation of Monte Carlo (MC) estimators are plotted, for increasing size of the sample set N. Each MC estimator simply takes the largest γ resulting from a bisection carried out for N samples of δ. The depicted empirical mean (solid line) and standard deviation (shaded area) are computed with respect to 100 MC estimators. The actual worst-case (at $\delta = -1$) and the worst-case out of 10^3 MC samples are plotted by horizontal dashed lines as a reference.

Figure 4.9 also shows approximations yielded by minimizing γ over the LFT projected Hamiltonian matrix. The expansion order is thereby adapted appropriately such that $N = L$. It is highlighted that equivalent results are obtained regardless if the approximation is computed by LFT Galerkin projection of the Hamiltonian or simply by taking the worst-case out of the L Gauss quadrature nodes for δ. Interestingly, the approximation by LFT Galerkin projection (Gaussian quadrature) can be observed to converge significantly faster — almost exponentially — compared to the Monte Carlo estimators, which converge rather slowly (asymptotically).

Figure 4.10 shows how the Gauss quadrature nodes "sample" the $\mathcal{H}_\infty$ norm $\|G(\delta)\|_\infty$. It can be seen how the "kink" in Figure 4.9 around $N = 1, ..., 4$ emerges, since for instance the $N = 1$ realization in Figure 4.10 is greater than the worst-case out of the

$N = 2$ realizations. In the present example, the results of the conventional Galerkin projection are not useful at all, since the projected system is always unstable. The corresponding approximations are therefore not shown in Figure 4.9.

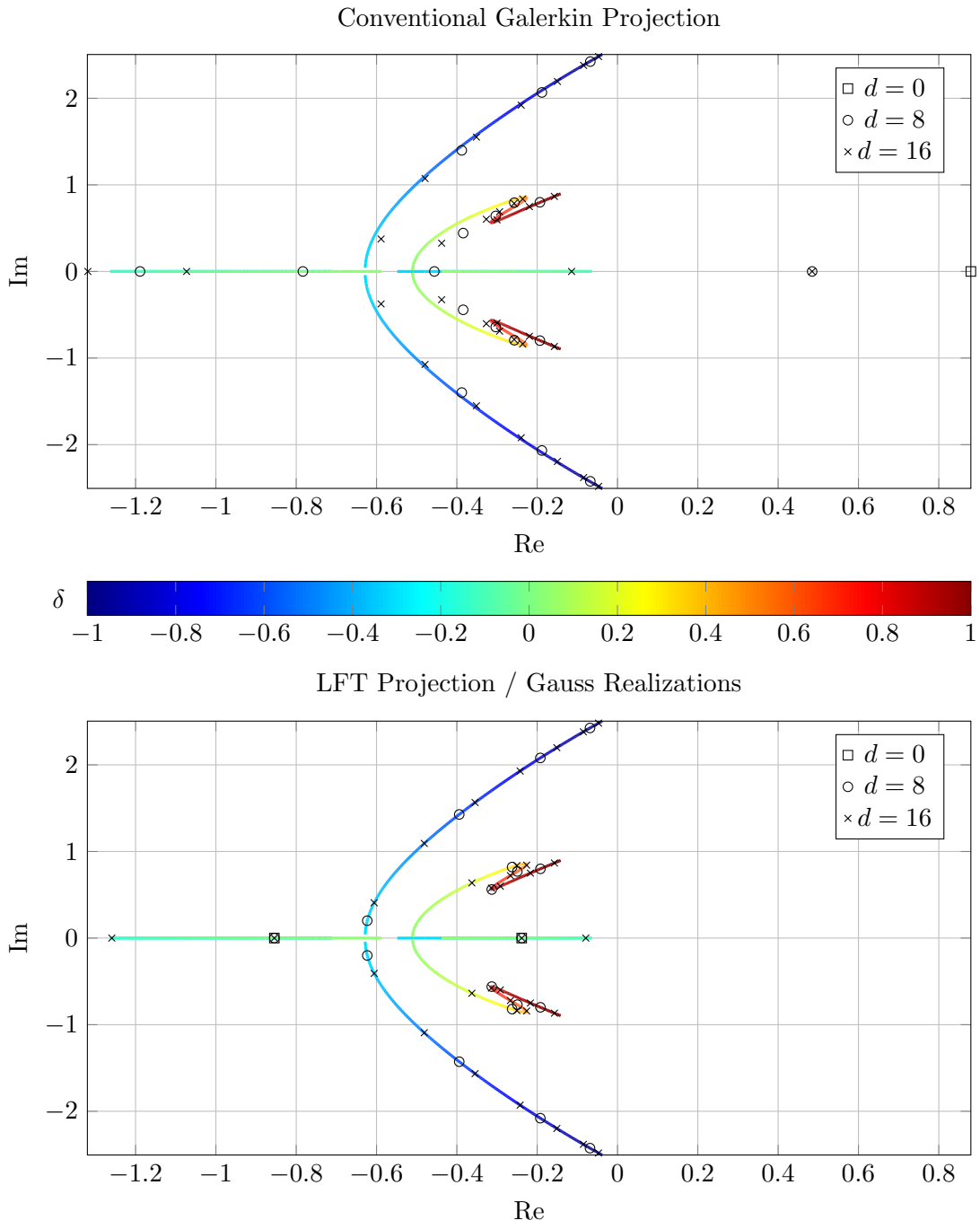


Figure 4.7: Eigenvalue comparison of the two Galerkin projection schemes: conventional approach (top) and projection of the system in LFT (bottom). The eigenvalues of $A(\delta)$ are plotted by the colored lines [28].

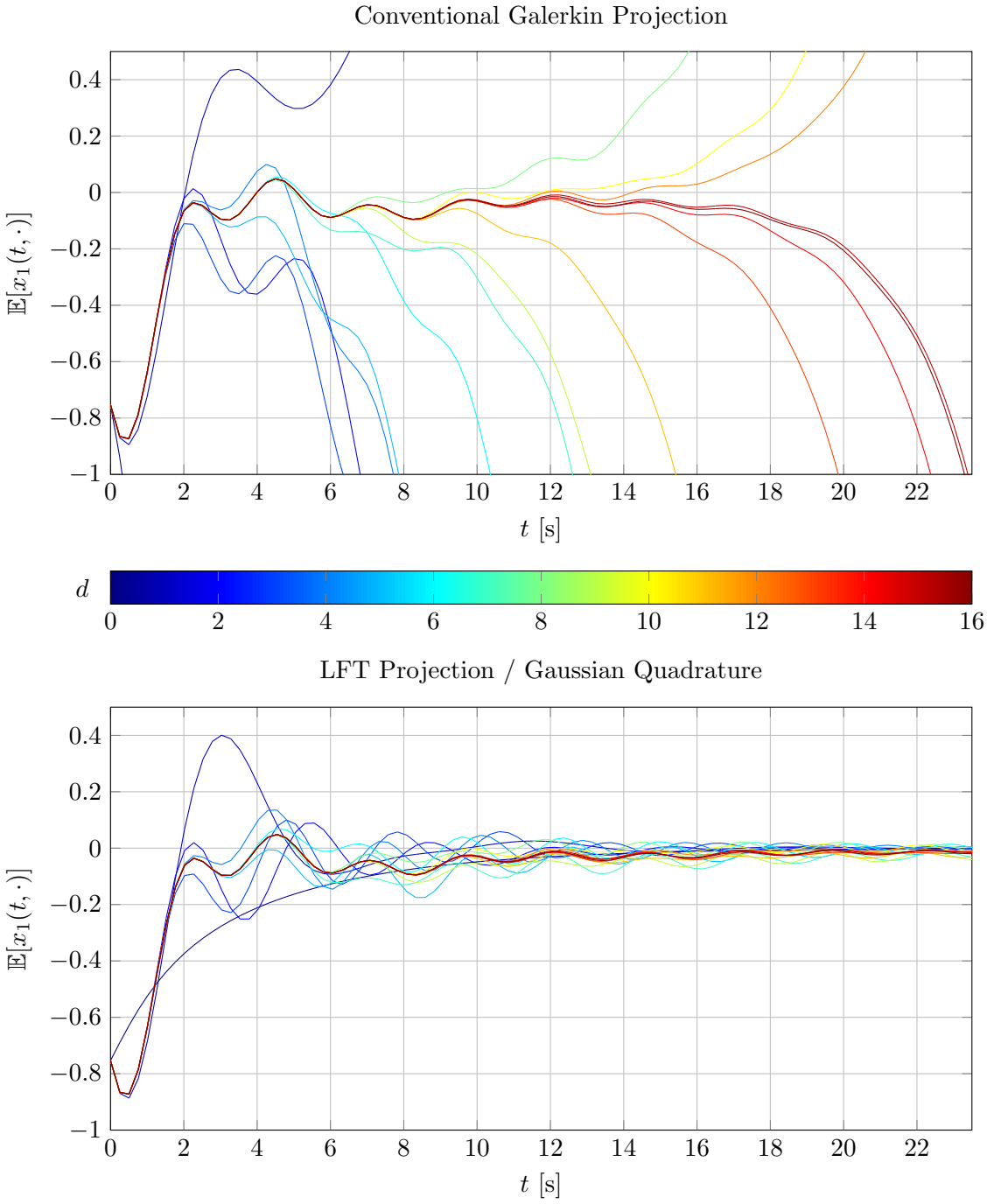


Figure 4.8: Time Response Comparison of Galerkin Projection Schemes [28]

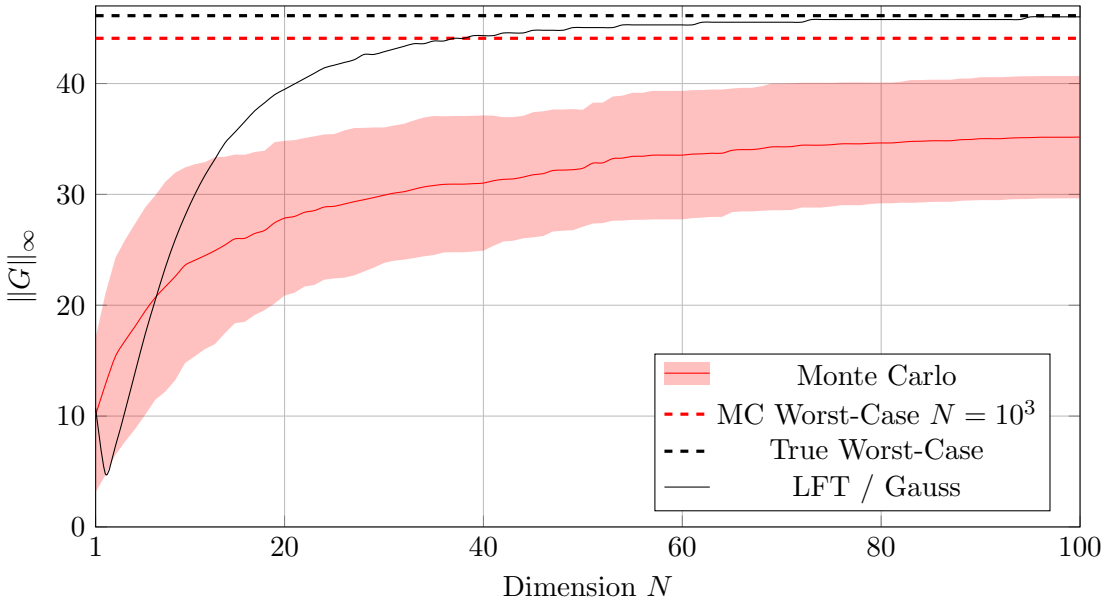


Figure 4.9: Different Approximations of the Robust $\mathcal{H}_\infty$-Norm [28]

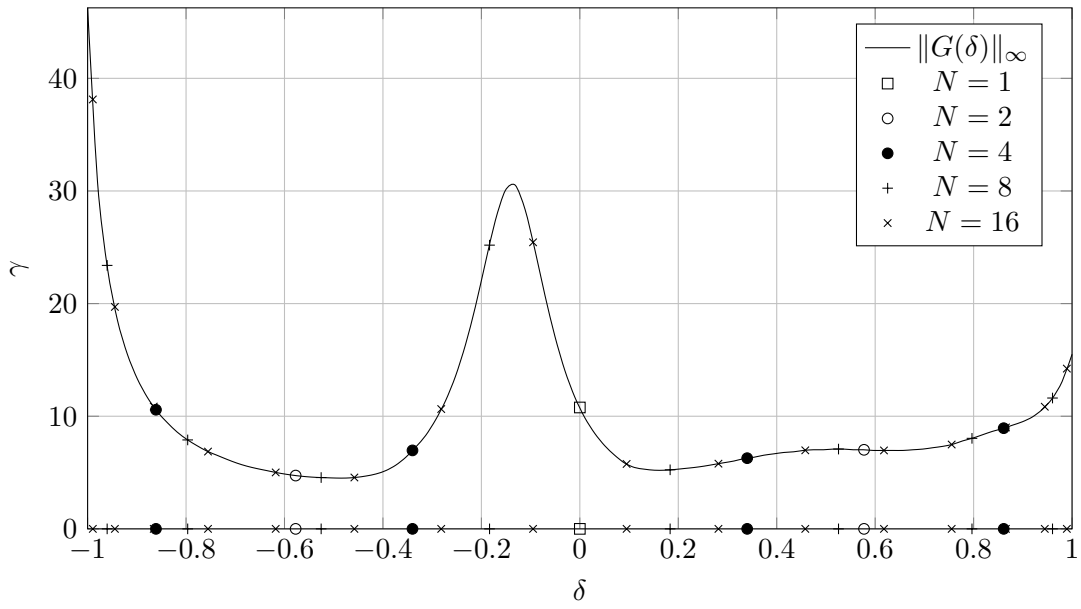


Figure 4.10: Estimation of the Robust $\mathcal{H}_\infty$-Norm

Truth... is much too complicated to allow anything but approximations.

John von Neumann

Chapter 5

RDE Projection

This chapter approximates the worst-case quadratic performance of uncertain linear systems defined over a finite time horizon. Recall that this robust performance analysis problem boils down to checking if the solution to a (random) parameter-dependent Riccati differential equation exists, see Chapter 2. Accordingly, the stage is set in Section 5.1. The parametric RDE is then approximately solved by a Galerkin projection based polynomial chaos series expansion. The approach is detailed in Section 5.2. It originates from the author's work in [26]. The effectiveness of the approximation is illustrated in Section 5.3 on the example of a worst-case analysis for a space launcher within the atmospheric ascent phase. The final two sections propose the LFT Galerkin projection approach for determining the projections of the RDE coefficient matrices. This approach is needed in the subsequent chapter. Section 5.5 discusses existence inheritance of the solution.

5.1 Riccati Differential Equations

Consider an uncertain linear time-varying system defined over a finite time horizon. According to Theorem 2.3, analyzing the robust quadratic performance of such systems boils down to certifying if the solution $X(t, \delta)$ to a Riccati differential equation

of the form

$$-\dot{X}(t,\delta) = \begin{bmatrix} I_{n_x} \\ X(t,\delta) \end{bmatrix}^T E(t,\delta) \begin{bmatrix} I_{n_x} \\ X(t,\delta) \end{bmatrix} \qquad (5.1)$$
$$X(T,\delta) = Q(T,\delta)$$

exists for all $(t,\delta) \in [0,T] \times \mathcal{D}$. Generally, *nominal* quadratic performance, i.e., for a fixed $\delta_0 \in \mathcal{D}$, is straightforward to evaluate simply by integrating the RDE numerically for $X(t,\delta_0)$. In practice, checking *robust* quadratic performance is computationally much harder, since existence of $X(t,\delta)$ needs to be certified $\forall \delta \in \mathcal{D}$. Since $\mathcal{D} \subset \mathbb{R}^{n_\delta}$, the RDE (5.1) is infinite-dimensional. To the best of the author's knowledge, there exist only approximate techniques to solve this problem (see Section 1.4). The main idea presented within this chapter is to apply a polynomial chaos series expansion to approximate the solution $X(t,\delta)$ and hence the worst-case quadratic performance of uncertain LTV systems.

The well-known smoothness properties of RDEs provide the justification and optimism for the applicability of this approach. It is known that the solutions of RDEs depend smoothly and vary monotonically with respect to the coefficient matrix $E(t,\delta)$ and the initial condition. Solutions $X(t,\delta)$ are smooth differentiable functions of δ, see [1] or [30] for details. Especially for smooth problems, polynomial chaos expansions converge well due to their spectral convergence behavior, see Theorem 3.2. The following section introduces our approach.

5.2 Galerkin Projection based PCE of Random RDEs

This section originates from the author's work in [26]. Assume a solution $X(t,\delta)$ to the random parameter-dependent RDE (5.1) exists for all $(t,\delta) \in [0,T] \times \mathcal{D}$. Solving for the strong solution $X(t,\delta)$ is computationally intractable, i.e., numerically integrating (5.1) exactly for all $\delta \in \mathcal{D}$. Instead, we seek a weak solution, determined by Galerkin projection (see Section 3.4) of the RDE. In essence, Galerkin projection approximates the

solution by a truncated series expansion

$$X(t,\delta) \approx Y(t,\delta) = \sum_{\alpha=0}^{L-1} Y_\alpha(t)\psi_\alpha(\delta). \tag{5.2}$$

The random parameter-dependent RDE (5.1) is projected on a suitable finite-dimensional orthogonal polynomial basis $\{\psi_\alpha \mid \alpha = 0, ..., L-1\}$. According to Section 3.4, the approximate solution is then determined by

$$\begin{aligned} \forall \alpha = 0, \ldots, L-1: \qquad -\langle \dot{Y}(t,\cdot), \psi_\alpha \rangle &= \left\langle \begin{bmatrix} I_{n_x} \\ Y(t,\cdot) \end{bmatrix}^T E(t,\cdot) \begin{bmatrix} I_{n_x} \\ Y(t,\cdot) \end{bmatrix},\ \psi_\alpha \right\rangle \\ \langle Y(T,\cdot), \psi_\alpha \rangle &= \langle Q(T,\cdot),\ \psi_\alpha \rangle. \end{aligned} \tag{5.3}$$

As usual, the inner products $\langle \cdot, \cdot \rangle$ are evaluated component-wise. Inserting (5.2) into (5.3), orthogonality of the polynomial basis yields

$$\begin{aligned} \dot{Y}_\alpha(t) &= -\frac{1}{\langle \psi_\alpha^2 \rangle} \left\langle \psi_\alpha,\ \begin{bmatrix} I_{n_x} \\ \sum_{\beta=0}^{L-1} Y_\beta(t)\psi_\beta \end{bmatrix}^T E(t,\cdot) \begin{bmatrix} I_{n_x} \\ \sum_{\xi=0}^{L-1} Y_\xi(t)\psi_\xi \end{bmatrix} \right\rangle \\ Y_\alpha(T) &= \frac{\langle Q(T,\cdot), \psi_\alpha \rangle}{\langle \psi_\alpha^2 \rangle} \qquad \forall \alpha = 0, \ldots, L-1. \end{aligned} \tag{5.4}$$

It is highlighted that Equation (5.4) is a coupled system of L deterministic RDEs. As is typical within the Galerkin approach, the projected system (5.4) has the same form as the original differential equation (5.1) (for deterministic realizations of δ). Equation (5.4) is coupled and of higher dimension increased by a factor of L. In order to make this point more clear, denote by E_{ij}, $i, j = 1, 2$, the $\mathbb{R}^{n_x \times n_x}$-valued blocks of the coefficient matrix $E(t,\delta)$

$$E(t,\delta) = \left[\begin{array}{c|c} E_{11}(t,\delta) & E_{12}(t,\delta) \\ \hline E_{21}(t,\delta) & E_{22}(t,\delta) \end{array}\right]. \tag{5.5}$$

Define by $\mathcal{E}(t,\delta)$ the symmetric $\mathbb{R}^{n_x(L+1)\times n_x(L+1)}$-valued matrix

$$\mathcal{E}(t,\delta) := \left[\begin{array}{c|ccc} E_{11} & E_{12}\psi_0 & \dots & E_{12}\psi_{L-1} \\ \hline E_{21}\psi_0 & E_{22}\psi_0\psi_0 & \dots & E_{22}\psi_0\psi_{L-1} \\ \vdots & \vdots & \ddots & \vdots \\ E_{21}\psi_{L-1} & E_{22}\psi_{L-1}\psi_0 & \dots & E_{22}\psi_{L-1}\psi_{L-1} \end{array}\right]. \tag{5.6}$$

Denote the Hilbert space projection of (5.6) onto ψ_α by

$$\mathcal{E}_\alpha(t) = \frac{1}{\langle\psi_\alpha^2\rangle}\langle\psi_\alpha, \mathcal{E}(t,\cdot)\rangle. \tag{5.7}$$

This allows us to write $\dot{Y}_\alpha(t)$ in Equation (5.4) as

$$\dot{Y}_\alpha(t) = -\begin{bmatrix} I_{n_x} \\ Y_0(t) \\ \vdots \\ Y_{L-1}(t) \end{bmatrix}^T \mathcal{E}_\alpha(t) \begin{bmatrix} I_{n_x} \\ Y_0(t) \\ \vdots \\ Y_{L-1}(t) \end{bmatrix} \quad \forall \alpha = 0,\dots,L-1. \tag{5.8}$$

This RDE clearly has the same form as the original differential equation (5.1), but is coupled and of higher dimension. In contrast to the computationally intractable, infinite-dimensional RDE (5.1), the approximate solution $Y(t,\delta)$ can be determined simply by numerically integrating the L coupled RDEs (5.8).

5.3 Launcher Analysis Example

This section demonstrates the effectiveness of the approximate worst-case analysis on the example of a launcher in atmospheric ascent. The analysis example is taken from [26]. It is also studied in [12] and originates from [80].

Consider the Vanguard space launcher, a launch vehicle from the early phase of space exploration. During its atmospheric ascent, significant perturbations from the nominal trajectory should be avoided in order to minimize the risk of the launcher being lost. Specifically, goal of the analysis is to quantify the effect of external wind disturbances

and parametric aerodynamic uncertainties on the launch trajectory. A linear time-varying model of the launcher's first-stage rigid body pitch dynamics is derived in [12] by linearizing with respect to a predefined gravity-turn trajectory:

$$\begin{bmatrix} \dot{\alpha}(t) \\ \dot{\theta}(t) \\ \dot{q}(t) \end{bmatrix} = \begin{bmatrix} \frac{Z_\alpha(t)}{m(t)v_d(t)} & \frac{-g\sin\theta_d(t)}{v_d(t)} & 1 \\ 0 & 0 & 1 \\ \frac{M_\alpha(t)}{J_{yy}(t)} & 0 & \frac{M_q(t)}{J_{yy}(t)} \end{bmatrix} \begin{bmatrix} \alpha(t) \\ \theta(t) \\ q(t) \end{bmatrix} + \begin{bmatrix} \frac{T_d}{m(t)v_d(t)} & \frac{Z_\alpha(t)}{m(t)v_d(t)} \\ 0 & 0 \\ \frac{T_d\xi}{J_{yy}(t)} & \frac{M_\alpha(t)}{J_{yy}(t)} \end{bmatrix} \begin{bmatrix} u(t) \\ d_\alpha(t) \end{bmatrix}. \tag{5.9}$$

Angle of attack α, pitch angle θ, and pitch rate q denote the states of the system. Input u is a corrective gimbal used for attitude control. The effect of wind is simulated by the disturbance input d_α perturbing the angle of attack. In [80], numerical values for the aerodynamic stability derivatives Z_α, M_α, and M_q are provided as functions of time. The time-varying mass m and pitch inertia J_{yy} are given analytically. The nominal values along the reference trajectory for speed v_d and pitch angle θ_d are given for $t \in [11.35\text{ s}, 146.35\text{ s}]$ with a step size of 2.7 s. Thrust T_d, gravitational acceleration g, and lever arm ξ of u with respect to the center of gravity are constants.

A time-invariant controller only using pitch angle feedback θ is provided in [80]. It consists of a linear quadratic regulator in combination with a full-order observer. The controller is designed in the region of maximum dynamic pressure. Additionally, a first-order actuator model with bandwidth 50 rad/s is included in the analysis. The closed-loop system matrix therefore has $n_x = 7$ states.

Due to inevitable modeling errors, assume the aerodynamic coefficients are uncertain

$$\delta = [Z_\alpha, M_\alpha, M_q]^T. \tag{5.10}$$

Each parameter δ_i is considered to be distributed uniformly around its nominal value $\bar{\delta}_i$ with $\delta_i \sim \mathcal{U}(0.75\bar{\delta}_i, 1.25\bar{\delta}_i)$. In order to quantify the effect of external wind disturbances on the uncertain closed-loop system, the finite-horizon induced $\mathcal{L}_2$-gain from $d_\alpha(t)$ to $\alpha(t)$ is to be computed. The finite-horizon induced $\mathcal{L}_2$-gain is

$$\|G\|_{\mathcal{L}_2[0,T]} = \sup \left\{ \frac{\|\alpha\|_{\mathcal{L}_2[0,T]}}{\|d_\alpha\|_{\mathcal{L}_2[0,T]}} \;\middle|\; d_\alpha \in \mathcal{L}_2[0,T] \setminus 0,\ x(0) = 0 \right\} = \gamma. \tag{5.11}$$

It falls into the class of quadratic performance metrics (see Chapter 2) by setting

$$\begin{aligned} Q(t,\delta) &= C^T(t,\delta)C(t,\delta),\ Q(T,\delta) = 0, \\ S(t,\delta) &= C^T(t,\delta)D(t,\delta), \\ R(t,\delta) &= D^T(t,\delta)D(t,\delta) - \gamma^2 I. \end{aligned} \tag{5.12}$$

A finite time horizon $t \in [15\text{ s},\ 100\text{ s}]$ is considered, marking the start and end point of the gravity-turn maneuver. For the following computations, a 0.5 s grid of the LTV model is generated by linear interpolation.

Approximations of the robust induced $\mathcal{L}_2$-gain given by the polynomial chaos series expansion $\hat{\gamma}_{PCE}$ and Monte Carlo (MC) sampling $\hat{\gamma}_{MC}$ are shown in Figure 5.1. The minimal γ for which an associated RDE solution exists is determined by a standard bisection algorithm. The worst-case out of $N = 10^4$ MC samples serves as a reference gain γ_{ref}. The nominal induced $\mathcal{L}_2$-gain of the system without parametric uncertainty is $\gamma_{nom} = 3.06$.

Using a total polynomial degree truncation scheme, five Galerkin projections (5.4) of the RDE (5.1) are computed. For this task, the *PolyChaos.jl* toolbox [51] within the Julia programming environment [7] is used. The smallest $\hat{\gamma}_{PCE}$ for which a solution to the RDE (5.4) exists is determined for each expansion degree $d \in \{0, 1, 2, 3, 4\}$. All involved Riccati differential equations are solved throughout using an order 5/4 explicit Runge-Kutta method with stiffness detection and automatic switching to `TRBDF2`, see [60].

The average $\hat{\gamma}_{MC}$ provided by 10 Monte Carlo estimators with sample size N is computed as a comparison. According to Section 3.3.2, applying a truncation scheme by total polynomial degree leads to $L = \frac{(n_\delta + d)!}{n_\delta! d!}$ terms in the expansion. Therefore, for each polynomial degree d, a MC estimator is calculated as the worst-case gain out of $N = L$ samples of δ. Empirical mean and standard deviation are calculated with respect to 10 MC estimators, indicating the randomness of the estimates. In contrast, the polynomial chaos approximation always yields a reproducible deterministic result.

It can be observed in Figure 5.1 that the polynomial chaos approximation converges to the reference worst-case from below. This convergence behavior (from below) could be reproduced for various other engineering application examples. However, it cannot be

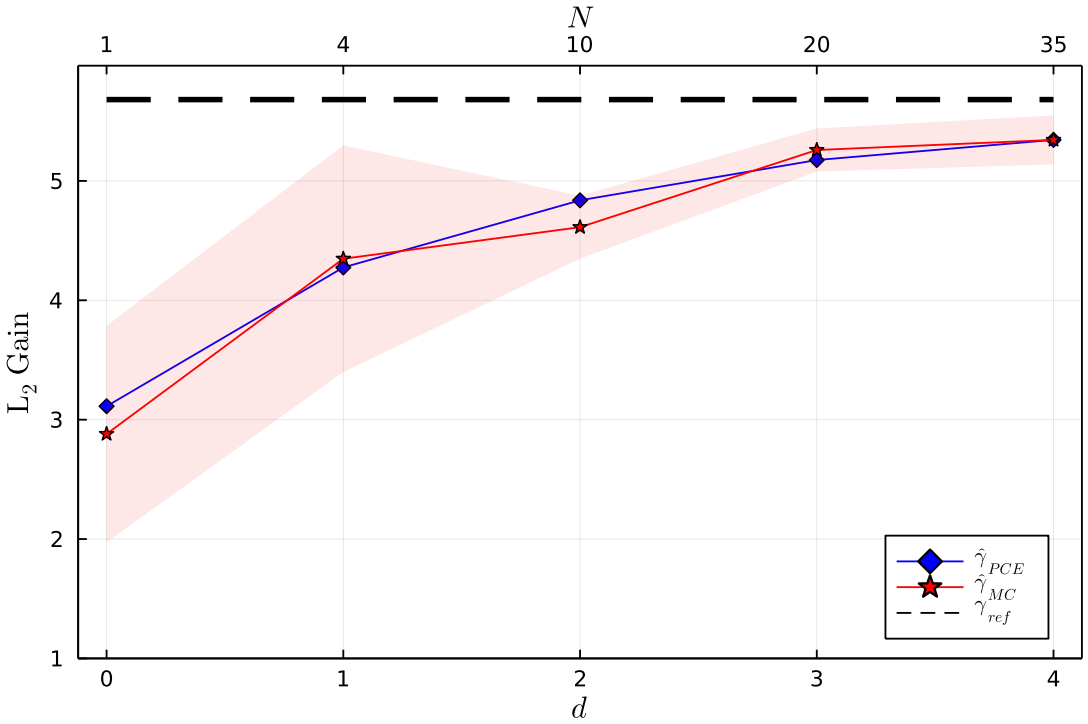


Figure 5.1: Approximated Robust Induced $\mathcal{L}_2$-Gain vs. PCE Degree d and MC Sample Set Dimension N [26]

generalized to yield a lower bound since there exist mathematical examples contradicting this convergence from below, see for instance the example given in Section 4.5.5.

In the following section, we introduce a method to determine the projection integrals of the RDE coefficient matrices given as a LFT. This method will be used in the subsequent chapter. In case the coefficient matrices are projected in this manner, the zeroth order PCE can in fact be shown to be a guaranteed lower bound to the true worst-case quadratic performance index, see Section 5.5.

5.4 LFT Projection of RDE Coefficient Matrices

Consider Equation (5.8) in Section 5.2. Therein, the projection integrals of the RDE coefficient matrix (5.5) need to be computed. Within this section, based on the work in [27], we propose to compute the required projection integrals $\langle \psi_\alpha, \cdot \rangle$, $\langle \psi_\alpha \psi_\beta, \cdot \rangle$, and

$\langle \psi_\alpha \psi_\beta \psi_\xi, \cdot \rangle$ by applying the LFT Galerkin projection approach elaborated in Chapter 4.

Following a similar notation as in Chapter 4, assume the LFT of the considered uncertain coefficient matrix is given by $\mathcal{F}_l\{M(t), \Delta(\delta)\}$, i.e.,

$$\begin{aligned} \begin{bmatrix} y(t,\delta) \\ v(t,\delta) \end{bmatrix} &= \begin{bmatrix} M_{11}(t) & M_{12}(t) \\ M_{21}(t) & M_{22}(t) \end{bmatrix} \begin{bmatrix} u(t,\delta) \\ w(t,\delta) \end{bmatrix} \\ w(t,\delta) &= \Delta(\delta) v(t,\delta). \end{aligned} \tag{5.13}$$

The expansion coefficients of $w(t,\delta)$ — now a matrix setting $u(t) = I_{n_u}$ in Figure 4.1 — can be concatenated within

$$W(t) = \Delta_\Pi \big[I - \big(M_{11}(t) \otimes I_L \big) \Delta_\Pi \big]^{-1} \big(M_{12}(t) \otimes e_1 \big) \tag{5.14}$$

by closing the expanded LFT. According to Chapter 4, the expansion coefficients $w_\alpha(t)$ are concatenated within the expanded matrix $W(t)$ as

$$W(t) = \big(I_{n_\Delta} \otimes \langle \Psi \Psi^T \rangle \big)^{-1} \langle w(t,\cdot) \otimes \Psi \rangle. \tag{5.15}$$

In the following, we switch from the Kronecker product notation to explicitly writing out the expansion sums for ease of exposition. Thus, the projection $\langle \psi_\alpha, y(t,\cdot) \rangle$ can be written as

$$\langle \psi_\alpha, \sum_{\beta=0}^{L-1} y_\beta(t) \psi_\beta \rangle = y_\alpha(t) \mathbb{E}[\psi_\alpha^2] \tag{5.16}$$

due to orthogonality of the basis polynomials ψ_α. Inserting the series expansion for $y(t,\delta)$ and $w(t,\delta)$ into

$$y(t,\delta) = M_{21}(t) w(t,\delta) + M_{22}(t) \tag{5.17}$$

it follows with (5.16)

$$y_\alpha(t) = M_{22}(t) \delta_{\alpha 0} + M_{21}(t) w_\alpha(t). \tag{5.18}$$

Higher-order projections can be obtained similarly, e.g.,

$$\langle \psi_\alpha \psi_\beta, \sum_{\xi=0}^{L-1} y_\xi(t) \psi_\xi \rangle = \langle \psi_\alpha \psi_\beta, M_{22}(t) \rangle + \langle \psi_\alpha \psi_\beta, \sum_{\xi=0}^{L-1} M_{21}(t) w_\xi(t) \psi_\xi \rangle \tag{5.19}$$

and

$$\langle \psi_\alpha \psi_\beta \psi_\xi, \sum_{\eta=0}^{L-1} y_\eta(t)\psi_\eta \rangle = \langle \psi_\alpha \psi_\beta \psi_\xi, M_{22}(t) \rangle + \langle \psi_\alpha \psi_\beta \psi_\xi, \sum_{\eta=0}^{L-1} M_{21}(t) w_\eta(t) \psi_\eta \rangle. \quad (5.20)$$

5.5 Existence Inheritance for $d = 0$

Assume the RDE coefficient matrices are projected as suggested within the previous section. In this case, the projected RDE (5.8) for a zeroth order expansion inherits existence of the solution from the original RDE (5.1). This can be deduced by the following reasoning.

For a polynomial degree $d = 0$ (i.e., $\alpha = 0$, $L = 1$), note that the projected matrix Δ_Π is obtained from $\Delta(\delta)$ simply by setting δ to its expected value:

$$W(t) = \mathbb{E}[\Delta] V(t) = \Delta\big(\mathbb{E}[\delta]\big) V(t). \quad (5.21)$$

It follows that the projected RDE (5.8) for Y_0 is given simply by the corresponding realization of (5.1) for the expected value of δ. Thus, the zeroth order PCE $Y_0(t)$ always exists provided the actual solution $X(t, \mathbb{E}[\delta])$ exists for all $t \in [0, T]$. We have therefore obtained a lower bound to the actual robust quadratic performance when applied in a bisection in order to find the minimal performance index γ for which the RDE exists.

At this point, we emphasize that existence of the zeroth order PCE is not inherited when applying the conventional Galerkin projection approach. This is due to the fact that, unfortunately, the expectation of a random matrix is not necessarily a realization of the original matrix. For example, albeit unlikely to encounter in practical applications, there exist system matrices $A(\delta)$ stable $\forall \delta \in \mathcal{D}$ such that the expectation $\mathbb{E}[A]$ (i.e., zeroth order PCE) is unstable, see Section 4.5.5. Therefore, existence of the zeroth order expanded RDE is not necessarily inherited when applying conventional Galerkin projection as in [26].

However, despite the absence of theoretical convergence guarantees, the approach still works well in practice. The situation is analogous to a similar problem of the original Fourier series: there also exist exemplary periodic functions for which convergence of the

original Fourier series expansion breaks down [42]. These mathematical counterexamples are unlikely to be encountered in real physical systems. This is illustrated by the following example.

Example 5.1. *Consider two system matrices for $\dot{x} = Ax$:*

$$A_1 = \begin{bmatrix} -\epsilon/2 & 1 \\ 0 & -\epsilon/2 \end{bmatrix}, \quad A_2 = \begin{bmatrix} -\epsilon/2 & 0 \\ 1 & -\epsilon/2 \end{bmatrix}, \tag{5.22}$$

with some small scalar $\epsilon > 0$. Assume the probability is 50% that the system is either given by A_1 or A_2. Note that the mathematical expectation is a weighted sum. This also applies to the stochastic modes of PCEs in general, which are defined via the respective projection relations [see Equation (3.27)]. Even though A_1 and A_2 are stable individually (both have two eigenvalues at $-\epsilon/2$), the sum $A_1 + A_2$ has two distinct eigenvalues, one at about -1 (stable) and one at $+1$ (unstable). This destabilizing phenomenon is similar to the example given in Section 4.5.5. However, the different structure of the matrices A_1 and A_2 implies that the order of integration switches, as becomes evident when drawing the block diagrams of both systems (consisting of two first-order systems in series). Physically, this means for instance in a mass-spring-damper system, the states for velocity and deflection from the equilibrium switch. This is unlikely to occur in practice.

Scientists investigate that which already is;
engineers create that which has never been.

Albert Einstein

Chapter 6

Automatic Landing System Analysis

Within this chapter, we illustrate the utility of the robustness analysis methods presented in the previous chapters. An industry-sized autolanding system of a civil transport aircraft is analyzed. The chapter is taken from the authors work in [27].

6.1 Autoland System Model

Throughout this chapter, the effect of parametric uncertainty and crosswind on the touchdown maneuver of an automatic landing system is analyzed. The aircraft system model is open-access available on `https://w3.onera.fr/smac/aircraftModel`. It was initiated within the autopilot design challenge [9] formulated by ONERA and Airbus. The model is representative of a civil transport aircraft in full configuration throughout the phases final approach and landing.

The aircraft equations of motion are implemented by a standard six-degrees-of-freedom flight mechanics model. The total sources of forces and moments acting on the airframe are: gravity, aerodynamics, and twin engines supplying equal thrust in the fuselage direction. The dynamic response of engines and control surface deflections (aileron, elevator, and rudder) are modeled by first-order actuator systems with rate and magnitude saturations.

The robust performance of the autopilot designed in [81] is to be analyzed in this

chapter. It is synthesized using loopshaping and robust control techniques. The control algorithm is split into the following common phases. In the first part of the final approach, the control objective is to let the aircraft track a desired glide path provided by the instrument landing system (ILS). Subsequently, the controller features two distinct switches, based on the measured height above ground. These switches initiate flare and decrab maneuvers. The switch to flare is at 20 m above ground, decrab at 5 m. The reader is referred to [81] for further details.

According to the original challenge formulation, the performance of the autoland system is evaluated with respect to six analysis criteria: short landing, long landing, hard landing, decentered landing, steep bank angle at touchdown, and steep wheel sideslip at touchdown. Since the goal of this chapter is to demonstrate the proposed analysis methods, we focus only on the hard landing requirement. In [81], this criterion is identified as the main performance limiting requirement. Of course, the analysis performed in the following sections can be repeated for the other criteria.

6.2 Disturbance Model

The hard landing requirement limits the aircraft's vertical sink rate at touchdown. Robust performance in the face of various sources of uncertainty, parameter variations, and atmospheric turbulence is to be assessed. As uncertain parameters, we consider throughout the total mass m, center of gravity location x^{CG}, as well as the aerodynamic coefficients C_L (lift) and C_M (pitching moment). Each parameter is distributed uniformly within the limits given in Table 6.1. Note that the center of gravity position is given relatively with respect to the mean aerodynamic chord. C_L and C_M are assumed to lie within 10% intervals of their mean values denoted by $\overline{C}_L$ and $\overline{C}_M$. The other parameters of the original problem formulation not listed in Table 6.1 are set to their mean values in the scope of this analysis. A complete reference of the original model parameter set can be found in [9].

Furthermore, the effect of static wind shear is added to the list of disturbances acting on the aircraft system model. Longitudinal head-/tailwind and lateral crosswind are considered. Denote by H^{LG} the landing gear's height above ground. Both wind profiles are implemented as quadratic functions of the altitude, with maxima at $H^{\mathrm{LG}} = 15$ m.

Table 6.1: Uncertain Parameters [27]

Parameter	min	max
Mass m	$120 \cdot 10^3$ kg	$180 \cdot 10^3$ kg
Center of Gravity x^{CG}	15 %	41 %
Lift Coefficient C_L	0.9 $\overline{C}_L$	1.1 $\overline{C}_L$
Pitch Coefficient C_M	0.9 $\overline{C}_m$	1.1 $\overline{C}_m$

Besides parametric uncertainty and static windshear, we include wind gusts and turbulence into the analysis. Atmospheric turbulence is considered according to [9] as first-order Markov processes, i.e., first-order filters driven by standard (unit variance) zero-mean white noise. For instance, at specific parameter values of 25 kts crosswind and 7.5 kts headwind, longitudinal turbulence is modeled by the transfer function

$$G_{w_u}(s) = \frac{20.139}{2.5s + 1} \tag{6.1}$$

excited by the white noise input mentioned earlier. Vertical turbulence is produced analogously with the wind filter

$$G_{w_w}(s) = \frac{1.7325}{2.5s + 1}. \tag{6.2}$$

Both transfer functions (6.1) and (6.2) have their output given in m/s.

6.3 Linear Time-Varying Model

An uncertain LTV system approximating the closed-loop aircraft dynamics is required in order to deploy the analysis methods proposed in the scope of this thesis. A dedicated LTV model is derived within this section. First, a nominal trajectory of the closed-loop system is obtained by executing the nonlinear simulation model. It is displayed in Figure 6.1. Starting point of the trajectory is 300 m above the runway, 30 m below the desired glide slope and 20 m left of the localizer signal. The time instants at which the controller switches to flare and decrab — in that order — are indicated by vertical

dash-dot lines. The gray thick line symbolizes the runway centerline. The landing gear's lateral displacement from the runway centerline is denoted by Y^{LG}. Hard landing is considered as a vertical sink rate of the landing gear $V_z^{\mathrm{LG}} \geq 12$ ft/s (3.7 m/s). Thus, the requirement avoiding a hard landing is specified as $V_z^{\mathrm{LG}} < 12$ ft/s. The constraint is visualized in the lower left plot by the horizontal red-dashed line. Moreover, the aircraft's orientation is shown in the lower right plot, as determined by the Euler angles Φ (bank), Θ (pitch), and Ψ (heading).

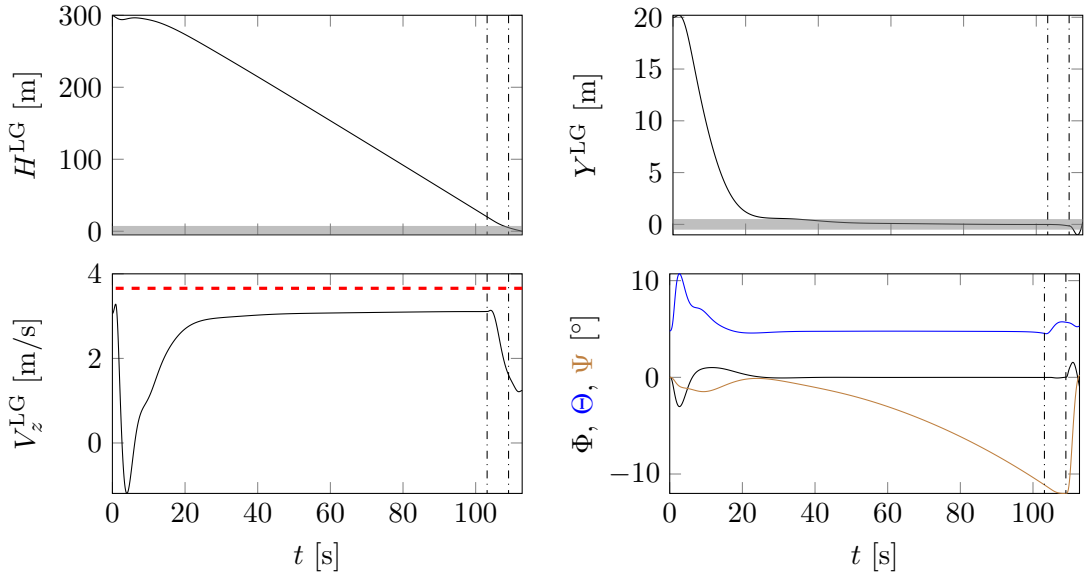


Figure 6.1: Nominal Trajectory without Turbulence [27]

All uncertain parameters are set to their mean values for the nominal trajectory. The mass is at 150 t, center of gravity is at 28%, headwind is 7.5 kts, etc. Merely the lateral crosswind value is fixed at its maximum: 25 kts. In order to obtain a smooth reference trajectory for the linearization performed subsequently, turbulence is not included within the nominal trajectory.

An uncertain LTV model is then generated as follows. Using Matlab & Simulink's `ulinearize` command, an uncertain linearization around the nominal trajectory is determined. In beforehand, parametric uncertainties are added to the original Simulink model by inserting `uncertain state-space` blocks. We highlight that the modeling within Simulink, use of `uncertain state-space` blocks, and the `ulinearize` command

are very intuitive and user-friendly, readily allowing an application of the proposed analysis methods to other engineering problems. Moreover, Matlab automatically generates the LTV model in linear fractional representation.

Inputs for the linearization are the white noise inputs of the turbulence filters (6.1) and (6.2). The nonlinear model is linearized at 20 equidistantly spaced points in time throughout the approach, flare, and decrab segments of the reference trajectory. The model is generated over the complete time interval by linear interpolation with respect to the linearization time points. Since the hard landing requirement limits the vertical sink rate, solely the longitudinal dynamics are considered. This implies sufficient decoupling from the lateral aircraft dynamics. Thus, the determined uncertain LTV model has $n_x = 20$ states.

6.4 Touchdown Analysis

This section checks satisfaction of the hard land requirement in face of the disturbances specified in Section 6.2. We apply the analysis methods of Chapters 4 and 5.

6.4.1 Trajectory Perturbations

First, we analyze deviations from the reference trajectory by considering mean and standard deviation of the linear perturbations. A polynomial chaos expansion of the uncertain LTV system is applied for this task. The expansion is computed for the system in linear fractional representation, see Section 4.3.

Let the white noise signals, i.e., inputs to the turbulence filters (6.1) and (6.2), be given. For a given input signal, the mean output due to parametric uncertainty of the LTV system is obtained by [see Equation (3.31)]

$$\mathbb{E}[y(t,\delta)] \approx \mathbb{E}\Big[\sum_{\alpha=0}^{L-1} y_\alpha(t)\psi_\alpha\Big] = \langle \psi_0, \sum_{\alpha=0}^{L-1} y_\alpha(t)\psi_\alpha\rangle = y_0(t). \tag{6.3}$$

The output variance is approximated similarly [cf. Equation (3.32)] by

$$\begin{aligned}\mathbb{V}[y(t,\delta)] &= \mathbb{E}\big[\big(y(t,\delta)-\mathbb{E}[y(t,\delta)]\big)\big(y(t,\delta)-\mathbb{E}[y(t,\delta)]\big)^T\big] \\ &\approx \big\langle \big(\sum_{\alpha=0}^{L-1} y_\alpha(t)\psi_\alpha - y_0(t)\big)\big(\sum_{\beta=0}^{L-1} y_\beta(t)\psi_\beta - y_0(t)\big)^T\big\rangle \\ &= \big\langle \sum_{\alpha=1}^{L-1} y_\alpha(t)\psi_\alpha \sum_{\beta=1}^{L-1} y_\beta^T(t)\psi_\beta\big\rangle \\ &= \sum_{\alpha=1}^{L-1} y_\alpha(t) y_\alpha^T(t)\langle\psi_\alpha^2\rangle.\end{aligned} \tag{6.4}$$

Note that Monte Carlo simulation (MCS) can simultaneously handle both the parametric uncertainty *and* the random input turbulence signals at once: for each parametric uncertainty sample, an associated random noise seed is determined, generating a unique white noise sequence. The empirical formulae for mean

$$\overline{y}(t) = \frac{1}{N}\sum_{i=1}^{N} y^{(i)}(t) \tag{6.5}$$

and (unbiased) variance

$$V_y(t) = \frac{1}{N-1}\sum_{i=1}^{N}\big(y^{(i)}(t)-\overline{y}(t)\big)\big(y^{(i)}(t)-\overline{y}(t)\big)^T \tag{6.6}$$

then yield the targeted variables of interest.

On the other hand, the method of polynomial chaos can account primarily for parametric uncertainty. In order to quantify the effect of the complete disturbance model detailed in Section 6.2 — including the random turbulence signal — the perturbations due to the white noise input and parametric uncertainty need to be combined. The approach applied for this task is summarized by pseudo-code given in Algorithm 1. The perturbation due to the white noise input is accounted for by Monte Carlo sampling, i.e., in a loop over the noise seeds, with N_{LTV} samples. For each noise seed, a specific input turbulence signal $u^{(i)}(t)$ is determined. Given the input signal, the perturbation due to parametric uncertainty is obtained by simulating the polynomial chaos expanded

Algorithm 1 Combined LFT expanded LTV Simulation and Monte Carlo Turbulence Sampling [27]

1: **for** $i = 1 : N_{LTV}$ **do**
2: Generate noise seed and input signal $u^{(i)}(t)$.
3: Simulate LFT expanded LTV system (4.17), (4.18).
4: Get mean $y_0^{(i)}(t)$ and standard deviation $\sigma_y^{(i)}(t)$ from (6.3), (6.4).
5: Set $\overline{y}_i(t) = \overline{y}_{i-1}(t) + \frac{y_0^{(i)}(t) - \overline{y}_{i-1}(t)}{i}$.
6: Get the combined standard deviation σ_i according to Equation (6.7).
7: **end for**

LTV system. In total, the combined mean is then still computed along the lines of Equation (6.5). Here, Welford's online algorithm (see [88]) is used for combining the mean of the first i noise seeds in Line 5. Line 6 computes the total standard deviation σ_i at iteration i by

$$\sigma_i^2 = \frac{(i-1)\left[\sigma_{i-1}^2 + (\overline{y}_{i-1} - \overline{y}_i)^2\right] + \left[\sigma_y^{(i)}\right]^2 + \left(y_0^{(i)} - \overline{y}_i\right)^2}{i}. \tag{6.7}$$

The reader is referred to Appendix A for detailed information on how to combine the standard deviation. Note that also a (Karhunen-Loeve) series expansion is applicable to the random input process (see e.g. [44]), as an alternative to the Monte Carlo loop. However, this is out of scope of the present thesis.

Throughout this chapter, the Galerkin projection of the uncertain system and its matrices are determined according to Sections 4.3 and 5.4, respectively. The uncertain system is given as a linear fractional transformation by Matlab's `ulinearize` command, see the previous section. Note that it is not straightforward to implement the conventional Galerkin approach by hand, since the system has $n_x = 20$ states and a non-trivial parameter dependence. This would therefore demand considerable programming effort and computational resources. At this point, the Galerkin projection of the system in linear fractional representation presents a viable and automated alternative.

Figure 6.2 displays the mean (solid line) and standard deviation (shaded area) computed by Algorithm 1 in blue. A polynomial expansion degree of 1 and $N_{LTV} = 100$ noise seeds are used. A Monte Carlo simulation of the full nonlinear model with $N_{MCS} = 10^4$ samples serves as a benchmark. It is shown in red. η denotes the elevator deflection.

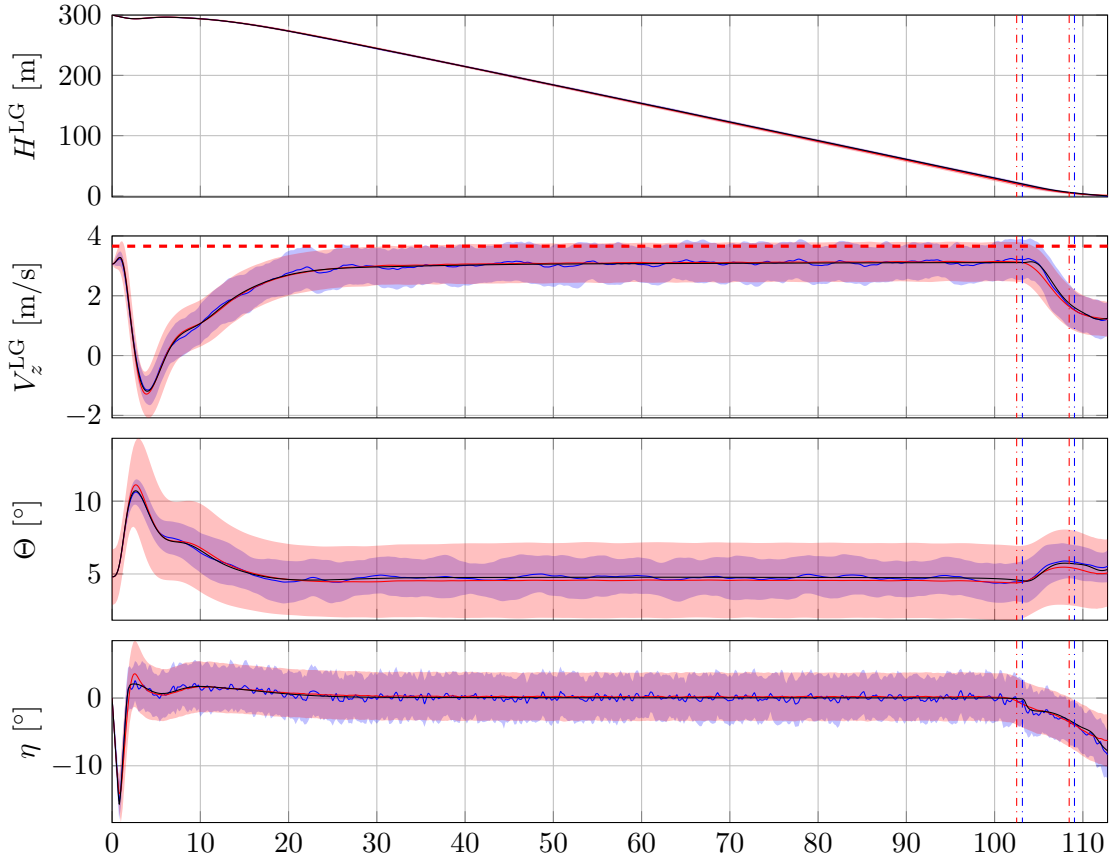


Figure 6.2: Comparison of Alg. 1 (blue) and Nonlinear MCS (red) [27]. Reference trajectory is in black.

The touchdown requirement for V_z^{LG} is shown by the horizontal red-dashed line. Vertical blue dash-dot lines mark the initiation of the flare and decrab maneuvers by the autoland controller. Regarding the LTV system analysis, these switching times are fixed by the reference trajectory. The mean time instants of when flare and decrab are initiated as determined by the MCS are indicated by vertical red dash-dot lines. A slight discrepancy is visible. On average, flare and decrab maneuvers are initiated sooner.

However, regarding especially the vertical sink rate V_z^{LG}, already a polynomial expansion degree of 1 yields a good approximation compared to the Monte Carlo (MC) reference with $N_{MCS} = 10^4$ samples. Moreover, on a standard laptop, the MCS requires $2 \cdot 10^4$ s (5.5 h) of computation time whereas Algorithm 1 with $N_{LTV} = 100$

simulations of the expanded LTV system requires 200 s (3.3 min) to execute. Hence, the time needed for a single simulation of the expanded LTV system is comparable to a single MC sample (about 2 s). Yet, for a fixed input signal, the PCE already accounts for the parametric uncertainty, i.e., it delivers the mean and standard deviation. This demonstrates that the PCE approach is beneficial especially when the analyzed system incorporates uncertain parameters.

Histograms of the vertical sink rate V_z^{LG} at touchdown are shown in Figure 6.3. The PCE approach (Algorithm 1 with $d = 1$ and $N_{LTV} = 100$) is displayed in blue and the MCS (with $N_{MCS} = 10^4$) in red. The histograms are scaled in the y-direction such that the bar areas add up to 1, imitating a probability distribution. The V_z^{LG} requirement is marked by the vertical red-dashed line. It can be observed that the MCS is biased towards harder touchdown speeds compared to Algorithm 1. A part of the bias can be traced back to be attributed to the earlier switching times to flare and decrab. This alters the reference trajectory. During the analysis, especially the switch to decrab was found to influence the touchdown conditions significantly. Apart from that, it has to be acknowledged that the linearization can only approximate the true system response.

6.4.2 Robust System Gain Analysis

An approximate worst-case analysis is also performed in the following. The estimated worst-case bounds for the vertical sink rate are added to Figure 6.3. These are determined as follows. We approximate the robust $\mathcal{L}_2$-to-Euclidean gain [see Equation (2.9)] by applying a polynomial chaos expansion along the lines of Chapter 5 for increasing total polynomial degrees d. Each series expansion is truncated by a standard total polynomial degree truncation scheme. The bounds plotted in Figure 6.3 are obtained in accordance with [14] by multiplying the approximated worst-case $\mathcal{L}_2$-to-Euclidean gain with a scaling factor and adding the reference sink rate at touchdown. Note that the approximate bounds tightly enclose all realizations obtained by the nonlinear MCS (and Algorithm 1).

We further consider more sophisticated truncation schemes in the following. Figure 6.4 displays different approximations of the worst-case gain. The robust $\mathcal{L}_2$-to-Euclidean gain $\|G\|_{2,\mathcal{L}_2}$ is plotted over the dimension N of the approximation used. For approximations based on polynomial chaos, N indicates the number of expansion terms.

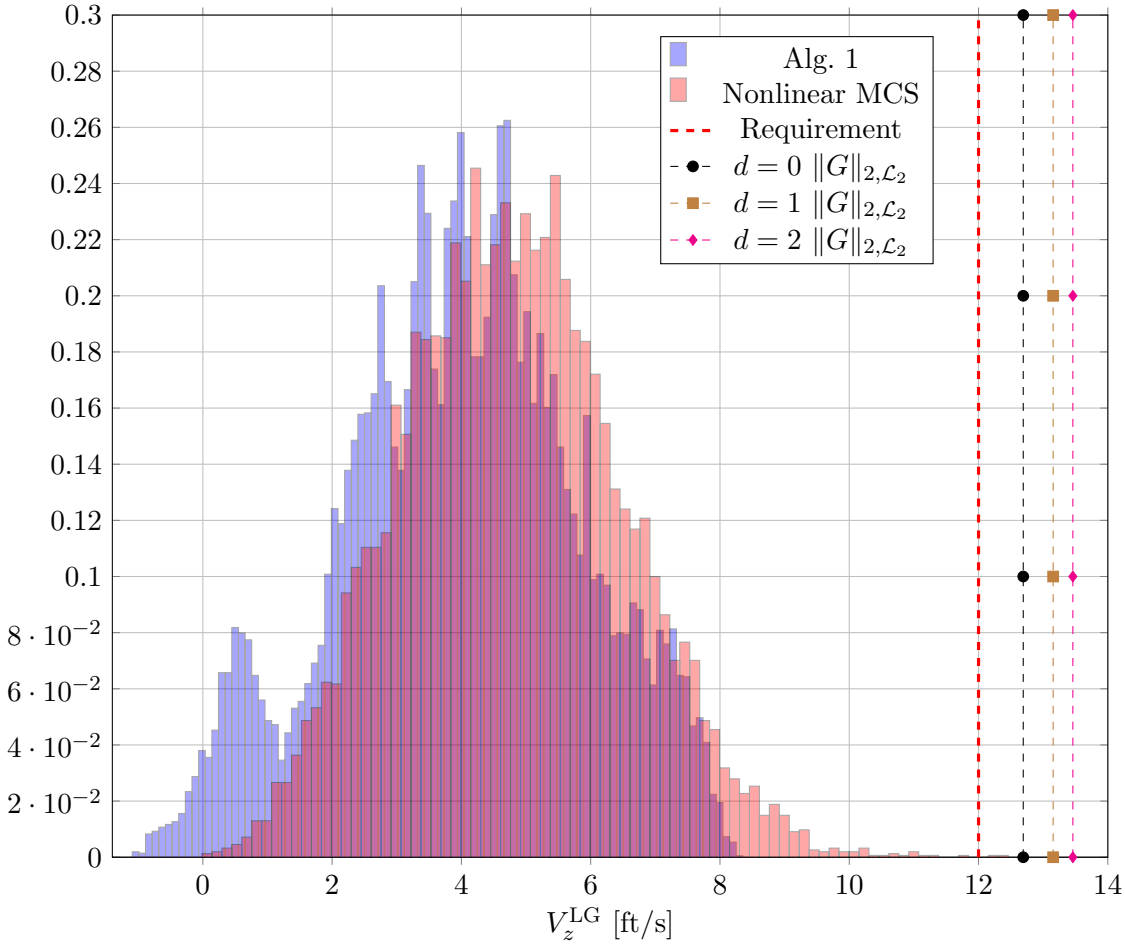


Figure 6.3: Vertical Sink Rate at Touchdown [27]

For Monte Carlo estimators, N is the size of the sample set. We emphasize that Monte Carlo is a *random* estimator. Every trial with N samples is a random variable. In contrast, the PCE always yields the same reproducible estimate (for a fixed polynomial basis). Regarding the Monte Carlo estimates, Figure 6.4 shows in red the empirical mean (solid line) and standard deviation (shaded area) with respect to 50 random estimators. The worst-case out of 10^3 Monte Carlo samples is plotted by the horizontal dashed line. The Matlab package LTVTools [70] is used for the involved bisection and solution of sampled instances of RDEs used by the MC estimators.

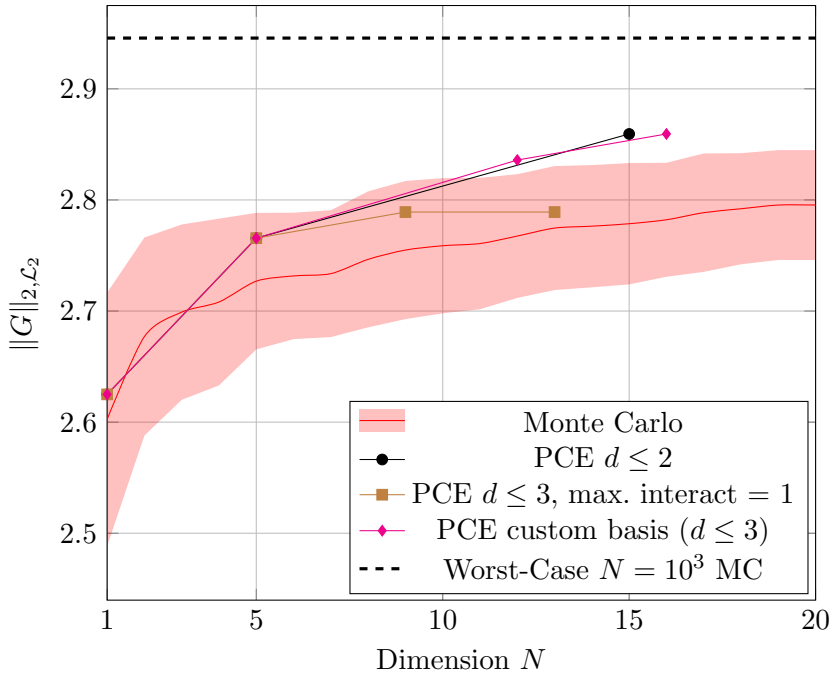


Figure 6.4: Approximate Robust $\mathcal{L}_2$-to-Euclidean Gains [27]

Besides the standard truncation scheme with total degree $d \leq 2$, a total degree truncation with $d \leq 3$ and maximum interaction limited to 1 as well as a custom adaptively built basis are used. See for instance [46] for details on advanced truncation schemes. In this example, they do not yield an advantage contrasted with the standard truncation scheme. Note that the zeroth ($N = 1$) and first-order ($N = 5$) expansion remain unaltered regardless which basis truncation scheme is used. The nominal $\mathcal{L}_2$-to-Euclidean gain is given by the first point at $N = 1$, since the zeroth order LFT expansion is given simply by the expectation of δ. The expectation matches the nominal case $\delta = 0$ for symmetric probability densities of δ.

The average gain computation time for one MC sample is 4.6 s. The total 10^3 samples need 4640 s (1.3 h) to compute. Table 6.2 logs the computation times needed by the polynomial chaos approximations employing the standard total degree truncation scheme. The right column shows the corresponding mean sample set size of a Monte Carlo estimator determined to fit the computation time of each row. Together with Figure 6.4, it is evident that the PCE yields quick and qualitatively good approximations

contrasted with the Monte Carlo estimators and the worst-case out of 10^3 samples. This has to be contrasted with the $N_{MCS} = 10^4$ nonlinear MCS in Figure 6.3, which needs 5.5 h computation time. Note that already the $d = 0$ gain approximation bounds all realizations of the MCS, including a realization at approximately 12.5 ft/s which violates the requirement.

Table 6.2: Polynomial Chaos Gain Approximation Times [27]

Total Degree d	Computation Time	Mean MC Sample Set Size N
0	5.6 s	1
1	73.8 s	16
2	1438.7 s	313

You live and learn. At any rate, you live.

The Hitchhiker's Guide to the Galaxy
Douglas Adams

Chapter 7

Conclusion and Discussion

7.1 Conclusions

Within this thesis, it is shown that a polynomial chaos expansion (PCE) of an uncertain linear time-varying (LTV) system in linear fractional representation (LFR) enables a computationally more efficient approach for probabilistic robustness analysis, see the first part of Chapter 4. In essence, the main advantage follows from exploiting the structure yielded by the uncertain system's linear fractional transformation (LFT) before deploying the PCE. The method's effectiveness is demonstrated on the example of a two-link robotic manipulator.

The second part of Chapter 4 shows that a PCE of an uncertain system in LFR is equivalent to applying Gaussian quadrature for numerical integration. This is interpreted in [28] as an argument to better apply Gaussian quadrature than a conventional polynomial chaos expansion based on Galerkin projection, for uncertain linear systems with rational parameter-dependence. We do not make this interpretation here. However, we do note that theoretically, conventional Galerkin projection may alter the differential equation in a manner such that important system characteristics, e.g., stability or existence of the solution, are not necessarily preserved. The LFT Galerkin projection is clearly advantageous here, since it yields realizations of the actual system. However, it is mentioned once more that there also exist exemplary periodic functions for which convergence of the original Fourier series breaks down as well [42]. Luckily, these situations seldom occur in real physical applications, see Example 5.1.

The equivalence of LFT Galerkin projection and Gaussian quadrature motivates to reconsider the problem we are actually solving throughout probabilistic robustness analysis, see the following section. In nearly all instances, the dominating challenge is computation of a probability integral. Practical solution approaches essentially boil down to means of efficient and accurate numerical integration. We have shown that applying PCE to uncertain systems in LFR essentially applies Gaussian quadrature. It is known that, for smooth (polynomial) integrands, Gaussian quadrature is the optimal approach and no other quadrature rule has a higher order of accuracy for a fixed number of nodes [77].

The previous theoretical findings are straightforward to apply to probabilistically robust controller synthesis, e.g., by a scenario approach [79] with realizations given by the Gauss quadrature points. Gaussian quadrature may further be applied to compute the probabilistic structured singular value (see for instance [5]).

Chapter 5 proposes to approximate the worst-case quadratic performance of uncertain LTV systems by applying a polynomial chaos expansion to the underlying Riccati differential equation (RDE). This Galerkin approximation can be applied throughout all fields of control systems theory where the solution of parametric RDEs, i.e., with random coefficients and initial conditions, is required. Existence inheritance of the solution is discussed in Section 5.5. Theoretically, this may be an issue, although unlikely to be encountered in practice. If simply the Gauss quadrature realizations are used, as suggested in Section 4.5.6, a lower bound to the true worst-case quadratic performance is obtained trivially. A lower bound is also obtained if the zeroth order PCE is determined by LFT Galerkin projection of the RDE coefficient matrices, see Section 5.5.

Finally, Chapter 6 demonstrates the usefulness of the methods proposed throughout this thesis on the example of a robustness analysis of an industrial-grade autolanding system. Therein, the proposed methods are shown to rapidly produce analysis results in good agreement with extensive Monte Carlo simulations. They can hence complement existing approaches for Verification & Validation, accelerate the control system design process, or provide a fast online robustness assessment (e.g., health monitoring) with reduced computational effort.

7.2 Further Remarks

As noted in the previous section, the ultimate undertaking within probabilistic robustness analysis is the computation of a probability integral. For nontrivial models and problems at the frontiers of engineering, the integral is not solvable analytically and must be approximated by numerical integration. However, the accurate and efficient numerical integration of a possibly high-dimensional integral is limited by our finite computational resources. Quadrature (i.e., gridding) suffers from the well-known curse of dimensionality. This explains the rapid growth of terms in the polynomial chaos expansion [see Equation (3.36)].

Arguably, Monte Carlo does not alleviate the curse of dimensionality. Rather, it is merely ignored. Nearly every application example given throughout this thesis shows that Monte Carlo estimators converge more slowly than a PCE or Gaussian quadrature with the same number of terms (i.e., samples). Instead of randomly fumbling around the parameter space in order to determine the nodes within a numerical quadrature rule, PCE and Gaussian quadrature make an informed choice based on the probability distribution of the input uncertainty δ.

Regarding the problem of scaling to high-dimensional parameter spaces, we remark the following. It is shown very intuitively in [6] that, with increasing dimension of the parameter space, the parameter regions which contribute most to the desired probability integral become more and more narrow. This problem amplifies to the point of singularity the greater the number of parameters. It is known that the application of any sound methodology is limited by the existence of the solution to the question asked [33]. One way to ensure a waste of time is by attempting to solve a problem with no solution. If the problem of a high-dimensional parameter domain persists, the method of Hamiltonian Monte Carlo [6] delivers promising results [93] worth further investigation and future research.

References

[1] H. Abou-Kandil, G. Freiling, V. Ionescu, and G. Jank, *Matrix Riccati Equations in Control and Systems Theory*, Birkhäuser, 2003. https://doi.org/10.1007/978-3-0348-8081-7.

[2] C. Audouze and P. Nair, *A Priori Error Analysis of Stochastic Galerkin Projection Schemes for Randomly Parametrized Ordinary Differential Equations*, International Journal for Uncertainty Quantification, 6 (2016), pp. 287–312. https://doi.org/10.1615/Int.J.UncertaintyQuantification.2016015843.

[3] F. Augustin and P. Rentrop, *Stochastic Galerkin techniques for random ordinary differential equations*, Numerische Mathematik, 122 (2012), pp. 399–419. https://doi.org/10.1007/s00211-012-0466-8.

[4] G. J. Balas, R. Y. Chiang, A. K. Packard, and M. G. Safonov, *Robust Control Toolbox, Version 3*, 2005. MathWorks, Natick, MA.

[5] G. J. Balas, P. J. Seiler, and A. K. Packard, *Analysis of an UAV Flight Control System Using Probabilistic μ*, in AIAA Guidance, Navigation, and Control Conf., 2012. https://doi.org/10.2514/6.2012-4989.

[6] M. Betancourt, *A Conceptual Introduction to Hamiltonian Monte Carlo*, 2018. https://doi.org/10.48550/arXiv.1701.02434.

[7] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, *Julia: A Fresh Approach to Numerical Computing*, SIAM Review, 59 (2017), pp. 65–98. https://doi.org/10.1137/141000671.

[8] R. Bhattacharya, *A Polynomial Chaos Framework for Designing Linear Parameter Varying Control Systems*, in Proc. American Control Conf., 2015, pp. 409–414. https://doi.org/10.1109/ACC.2015.7170770.

[9] J.-M. Biannic and J. Boada-Bauxell, *A Civilian Aircraft Landing Challenge*, 2016. On-line available from the aerospace benchmark section of the SMAC Toolbox. http://w3.onera.fr/smac/.

[10] J.-M. Biannic, C. Roos, S. Bennani, F. Boquet, V. Preda, and B. Girouart, *Advanced probabilistic μ-analysis techniques for AOCS validation*, European Journal of Control, 62 (2021), pp. 120–129. https://doi.org/10.1016/j.ejcon.2021.06.019.

[11] F. Biertümpfel, *Robustness Analysis of Linear Time-Varying Systems with Application to Aerospace Systems*, PhD thesis, University of Nottingham, 2021.

[12] F. Biertümpfel and H. Pfifer, *Worst Case Gain Computation of Linear Time-Varying Systems over a Finite Horizon*, in Proc. IEEE Conf. Control Technology and Applications, 2018, pp. 952–957. https://doi.org/10.1109/CCTA.2018.8511591.

[13] F. Biertümpfel and H. Pfifer, *Finite Horizon Touchdown Analysis of Autolanded Aircraft under Crosswind*, IFAC-PapersOnLine, 54 (2021), pp. 124–129. https://doi.org/10.1016/j.ifacol.2021.08.591.

[14] F. Biertümpfel and H. Pfifer, *Finite horizon analysis of autolanded aircraft in final approach under crosswind*, Control Engineering Practice, 122 (2022), p. 105105. https://doi.org/10.1016/j.conengprac.2022.105105.

[15] F. Biertümpfel, H. Pfifer, and S. Bennani, *Finite Horizon Worst Case Analysis of Launch Vehicles*, in 21st IFAC Symposium on Automatic Control in Aerospace, 2019, pp. 31–36. https://doi.org/10.1016/j.ifacol.2019.11.065.

[16] V. D. Blondel and J. N. Tsitsiklis, *A survey of computational complexity results in systems and control*, Automatica, 36 (2000), pp. 1249–1274. https://doi.org/10.1016/S0005-1098(00)00050-9.

[17] H. Bode, *Variable equalizers*, Bell System Technical Journal, (1938), pp. 229–244.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004. https://doi.org/10.1017/CBO9780511804441.

[19] J. R. Buch, *Finite Horizon Robustness with Applications to Missile Engagements*, PhD thesis, University of Minnesota, 2021.

[20] G. C. Calafiore, F. Dabbene, and R. Tempo, *Research on probabilistic methods for control system design*, Automatica, 47 (2011), pp. 1279–1293. https://doi.org/10.1016/j.automatica.2011.02.029.

[21] R. H. Cameron and W. T. Martin, *The Orthogonal Development of Non-Linear Functionals in Series of Fourier-Hermite Functionals*, Annals of Mathematics, 48 (1947), pp. 385–392. https://doi.org/10.2307/1969178.

[22] J. Doyle, *Guaranteed Margins for LQG Regulators*, IEEE Trans. Automatic Control, 23 (1978), pp. 756–757. https://doi.org/10.1109/TAC.1978.1101812.

[23] J. Doyle, *Analysis of feedback systems with structured uncertainties*, IEE Proceedings D (Control Theory and Applications), 129 (1982), pp. 242–250. https://doi.org/10.1049/ip-d.1982.0053.

[24] O. G. Ernst, A. Mugler, H.-J. Starkloff, and E. Ullmann, *On the convergence of generalized polynomial chaos expansions*, ESAIM: Mathematical Modelling and Numerical Analysis, 46 (2012), pp. 317–339. https://doi.org/10.1051/m2an/2011045.

[25] L. L. Evangelisti and H. Pfifer, *Probabilistic Robustness Analysis of Uncertain LTV Systems in Linear Fractional Representation*, IEEE Control Systems Letters, 6 (2021), pp. 428–433. https://doi.org/10.1109/LCSYS.2021.3078881.

[26] L. L. Evangelisti and H. Pfifer, *Polynomial Chaos Approximation of the Quadratic Performance of Uncertain Time-Varying Linear Systems*, in Proc. American Control Conf., 2022, pp. 1853–1858. https://doi.org/10.23919/ACC53348.2022.9867871.

[27] L. L. Evangelisti and H. Pfifer, *Finite Horizon Robustness Analysis of an Automatic Landing System under Probabilistic Uncertainty*, Journal of Guidance, Control, and Dynamics, (2023). submitted for publication.

[28] L. L. Evangelisti and H. Pfifer, *On the Application of Galerkin Projection based Polynomial Chaos in Linear Systems and Control*, Automatica, (2023). submitted for publication.

[29] J. Fisher and R. Bhattacharya, *Linear quadratic regulation of systems with stochastic parameter uncertainties*, Automatica, 45 (2009), pp. 2831–2841. https://doi.org/10.1016/j.automatica.2009.10.001.

[30] G. Freiling and V. Ionescu, *Monotonicity and convexity properties of matrix Riccati equations*, IMA Journal of Mathematical Control and Information, 18 (2001), pp. 61–72. https://doi.org/10.1093/imamci/18.1.61.

[31] W. Gautschi, *Orthogonal Polynomials: Computation and Approximation*, Oxford University Press, 2004. https://doi.org/10.1093/oso/9780198506720.001.0001.

[32] M. Gerritsma, J.-B. van der Steen, P. Vos, and G. Karniadakis, *Time-dependent generalized polynomial chaos*, Journal of Computational Physics, 229 (2010), pp. 8333–8363. https://doi.org/10.1016/j.jcp.2010.07.020.

[33] M. Green and D. J. Limebeer, *Linear Robust Control*, Prentice Hall, 1995.

[34] L. P. Hansen and T. J. Sargent, *Robustness*, Princeton University Press, 2008.

[35] S. Hecker, *Generation of low order LFT Representations for Robust Control Applications*, PhD thesis, Technical University Munich, 2006.

[36] S. Hecker, A. Varga, and J.-F. Magni, *Enhanced LFR-toolbox for Matlab and LFT-based gain scheduling*, in Proc. 6th ONERA-DLR Aerospace Symposium, 2004, pp. 80–89.

[37] F. S. Hover and M. S. Triantafyllou, *Application of polynomial chaos in stability and control*, Automatica, 42 (2006), pp. 789–795. https://doi.org/10.1016/j.automatica.2006.01.010.

[38] R. A. Hyde, *The Application of Robust Control to VSTOL Aircraft*, PhD thesis, University of Cambridge, 1991.

[39] A. Ianelli, P. Seiler, and A. Marcos, *Worst-Case Disturbances for Time-Varying Systems with Application to Flexible Aircraft*, Journal of Guidance, Control, and Dynamics, 42 (2019), pp. 1261–1271. https://doi.org/10.2514/1.G004023.

[40] H.-D. Joos, *Application of Optimization-Based Worst Case Analysis to Control Law Assessment in Aerospace*, in Advances in Aerospace Guidance, Navigation and Control, J. Bordeneuve-Guibé, A. Drouin, and C. Roos, eds., Springer, 2015, pp. 53–65. https://doi.org/10.1007/978-3-319-17518-8_4.

[41] K.-K. K. Kim, D. E. Shen, Z. K. Nagy, and R. D. Braatz, *Wiener's Polynomial Chaos for the Analysis and Control of Nonlinear Dynamical Systems with Probabilistic Uncertainties [Historical Perspectives]*, IEEE Control Systems Magazine, 33 (2013), pp. 58–67. https://doi.org/10.1109/MCS.2013.2270410.

[42] A. Kolmogoroff, *Une série de Fourier-Lebesgue divergente presque partout*, Fundamenta Mathematicae, 4 (1923), pp. 324–328.

[43] O. P. Le Maître and O. M. Knio, *Spectral Methods for Uncertainty Quantification*, Springer Dordrecht, 2010. https://doi.org/10.1007/978-90-481-3520-2.

[44] D. Lucor, C.-H. Su, and G. E. Karniadakis, *Generalized polynomial chaos and random oscillators*, International Journal for Numerical Methods in Engineering, 60 (2004), pp. 571–596. https://doi.org/10.1002/nme.976.

[45] J.-F. Magni, S. Bennani, and J. Terlouw, eds., *Robust Flight Control: A Design Challenge*, Springer, 1997. https://doi.org/10.1007/BFb0113842.

[46] S. Marelli, N. Lüthen, and B. Sudret, *UQLab User Manual – Polynomial Chaos Expansions*, 2022. Technical report UQLab-V2.0-104. Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland.

[47] B. S. May, *Probabilistic Robust Control: Theory and Applications*, PhD thesis, California Institute of Technology, 1997.

[48] A. Megretski and A. Rantzer, *System Analysis via Integral Quadratic Constraints*, IEEE Trans. Automatic Control, 42 (1997), pp. 819–830. https://doi.org/10.1109/9.587335.

[49] T. Mühlpfordt, *Uncertainty Quantification via Polynomial Chaos Expansion – Methods and Applications for Optimization of Power Systems*, PhD thesis, Karlsruher Institut für Technologie (KIT), 2019.

[50] T. Mühlpfordt, R. Findeisen, V. Hagenmeyer, and T. Faulwasser, *Comments on Truncation Errors for Polynomial Chaos Expansions*, IEEE Control Systems Letters, 2 (2018), pp. 169–174. https://doi.org/10.1109/LCSYS.2017.2778138.

[51] T. Mühlpfordt, F. Zahn, V. Hagenmeyer, and T. Faulwasser, *PolyChaos.jl – A Julia Package for Polynomial Chaos in Systems and Control*, in Proc. 21st IFAC World Congress, 2020, pp. 7210–7216. https://doi.org/10.1016/j.ifacol.2020.12.552.

[52] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, 1994. https://doi.org/10.1201/9781315136370.

[53] Z. K. Nagy and R. D. Braatz, *Distributional uncertainty analysis using power series and polynomial chaos expansions*, Journal of Process Control, 17 (2007), pp. 229–240. https://doi.org/10.1016/j.jprocont.2006.10.008.

[54] D. Ossmann, J. Theis, and P. Seiler, *Robust Control Design for Load Reduction on a Liberty Wind Turbine*, in Proc. ASME Dynamic Systems and Control Conf., 2016. https://doi.org/10.1115/DSCC2016-9719.

[55] J. A. Paulson, A. Mesbah, S. Streif, R. Findeisen, and R. D. Braatz, *Fast Stochastic Model Predictive Control of High-dimensional Systems*, in Proc. 53rd IEEE Conf. Decision and Control, 2014, pp. 2802–2809. https://doi.org/10.1109/CDC.2014.7039819.

[56] I. R. Petersen and R. Tempo, *Robust control of uncertain systems: Classical results and recent developments*, Automatica, 50 (2014), pp. 1315–1335. https://doi.org/10.1016/j.automatica.2014.02.042.

[57] H. Pfifer, *LPV/LFT Modeling and its Application in Aerospace*, PhD thesis, Technische Universität München, 2013.

[58] I. Postlethwaite, N. P. Foster, and D. J. Walker, *Rotorcraft control law design for rejection of atmospheric turbulence*, in International Conf. on Control, 1994, pp. 1284–1289. https://doi.org/10.1049/cp:19940322.

[59] R. Pulch and F. Augustin, *Stability Preservation in Stochastic Galerkin Projections of Dynamical Systems*, SIAM/ASA Journal on Uncertainty Quantification, 7 (2019), pp. 634–651. https://doi.org/10.1137/17M1142223.

[60] C. Rackauckas and Q. Nie, *DifferentialEquations.jl – A Performant and Feature-Rich Ecosystem for Solving Differential Equations in Julia*, Journal of Open Research Software, 5 (2017), p. 15. https://doi.org/10.5334/jors.151.

[61] C. Rohrs, L. Valavani, and M. Athans, *Convergence studies of adaptive control algorithms part I: Analysis*, in Proc. 19th IEEE Conf. Decision and Control, 1980, pp. 1138 – 1141. https://doi.org/10.1109/CDC.1980.271981.

[62] L. E. Ryan and R. F. Stengel, *Stochastic Stability and Performance Robustness of Linear Multivariable Systems*, in Proc. American Control Conf., 1990, pp. 462–468. https://doi.org/10.23919/ACC.1990.4790779.

[63] M. G. Safonov, *Stability margins of diagonally perturbed multivariable feedback systems*, IEE Proceedings D (Control Theory and Applications), 129 (1982), pp. 251–256. https://doi.org/10.1049/ip-d.1982.0054.

[64] M. G. Safonov, *Origins of Robust Control: Early History and Future Speculations*, in 7th IFAC Symposium on Robust Control Design, 2012, pp. 1–8. https://doi.org/10.3182/20120620-3-DK-2025.00179.

[65] M. G. Safonov and R. Y. Chiang, *CACSD Using the State-Space L^∞ Theory – A Design Example*, IEEE Trans. Automatic Control, 33 (1988), pp. 477–479. https://doi.org/10.1109/9.1232.

[66] F. Saupe, *Linear Parameter Varying Control Design for Industrial Manipulators*, PhD thesis, Technische Universität Hamburg, 2013.

[67] C. Scherer, P. Gahinet, and M. Chilali, *Multiobjective Output-Feedback Control via LMI Optimization*, IEEE Trans. Automatic Control, 42 (1997), pp. 896–911. https://doi.org/10.1109/9.599969.

[68] C. Scherer and S. Weiland, *Linear Matrix Inequalities in Control*, 2015. Lecture Notes.

[69] K. Schweidel, J. Buch, P. Seiler, and M. Arcak, *Computing Worst-Case Disturbances for Finite-Horizon Linear Time-Varying Approximations of Uncertain Systems*, IEEE Control Systems Letters, 5 (2021), pp. 1753–1758. https://doi.org/10.1109/LCSYS.2020.3043843.

[70] P. Seiler, J. Buch, R. M. Moore, C. Meissen, M. Arcak, and A. Packard, *LTVTools (Beta), A Matlab Toolbox for Linear Time-Varying Systems*, 2021. https://github.com/buchjyot/LTVTools.

[71] P. Seiler, R. Moore, C. Meissen, M. Arcak, and A. Packard, *Finite horizon robustness analysis of LTV systems using integral quadratic constraints*, Automatica, 100 (2019), pp. 135–143. https://doi.org/10.1016/j.automatica.2018.11.009.

[72] J. S. Shamma, *Analysis and Design of Gain Scheduled Control Systems*, PhD thesis, Massachusetts Institute of Technology, 1988.

[73] D. E. Shen, S. Lucia, Y. Wan, R. Findeisen, and R. D. Braatz, *Polynomial Chaos-Based $\mathcal{H}_2$-optimal Static Output Feedback Control of Systems with Probabilistic Parametric Uncertainties*, in Proc. 20th IFAC World Congress, 2017, pp. 3536–3541. https://doi.org/10.1016/j.ifacol.2017.08.949.

[74] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control*, John Wiley & Sons, 2005.

[75] J. Sofrony, M. Turner, and I. Postlethwaite, *Anti-Windup Synthesis Using Riccati Equations*, in Proc. 16th IFAC World Congress, 2005, pp. 171–176. https://doi.org/10.3182/20050703-6-CZ-1902.00599.

[76] B. E. Sonday, R. D. Berry, H. N. Najm, and B. J. Debusschere, *Eigenvalues of the Jacobian of a Galerkin-Projected Uncertain ODE System*, SIAM Journal on Scientific Computing, 33 (2011), pp. 1212–1233. https://doi.org/10.1137/100785922.

[77] T. J. Sullivan, *Introduction to Uncertainty Quantification*, Springer Cham, 2015. https://doi.org/10.1007/978-3-319-23395-6.

[78] G. Tadmor, *Worst-Case Design in the Time Domain: The Maximum Principle and the Standard H_∞ Problem*, Mathematics of Control, Signals, and Systems, 3 (1990), pp. 301–324. https://doi.org/10.1007/BF02551373.

[79] R. Tempo, G. Calafiore, and F. Dabbene, *Randomized Algorithms for Analysis and Control of Uncertain Systems*, Springer London, 2013. https://doi.org/10.1007/978-1-4471-4610-0.

[80] A. Tewari, *Automatic Control of Atmospheric and Space Flight Vehicles: Design and Analysis with MATLAB and Simulink*, Birkhäuser, 2011. https://doi.org/10.1007/978-0-8176-4864-0.

[81] J. Theis, D. Ossmann, F. Thielecke, and H. Pfifer, *Robust autopilot design for landing a large civil aircraft in crosswind*, Control Engineering Practice, 76 (2018), pp. 54–64. https://doi.org/10.1016/j.conengprac.2018.04.010.

[82] J. Theis, H. Pfifer, and P. J. Seiler, *Robust Control Design for Active Flutter Suppression*, in AIAA Atmospheric Flight Mechanics Conf., 2016. https://doi.org/10.2514/6.2016-1751.

[83] G. Vinnicombe, *Measuring the Robustness of Feedback Systems*, PhD thesis, University of Cambridge, 1993.

[84] G. Vinnicombe, *Uncertainty and Feedback*, Imperial College Press, 2001.

[85] Y. Wan and R. D. Braatz, *Mixed Polynomial Chaos and Worst-Case Synthesis Approach to Robust Observer based Linear Quadratic Regulation*, in Proc. American Control Conf., 2018, pp. 6798–6803. https://doi.org/10.23919/ACC.2018.8431698.

[86] Y. Wan, D. E. Shen, S. Lucia, R. Findeisen, and R. D. Braatz, *A Polynomial Chaos Approach to Robust $\mathcal{H}_\infty$ Static Output-Feedback Control With Bounded Truncation Error*, IEEE Trans. Automatic Control, 68 (2023), pp. 470–477. https://doi.org/10.1109/TAC.2022.3140275.

[87] K. Weierstrass, *Über die analytische Darstellbarkeit sogenannter willkürlicher Functionen einer reellen Veränderlichen*, in Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften zu Berlin, 1885, pp. 633–639, 789–805.

[88] B. P. Welford, *Note on a Method for Calculating Corrected Sums of Squares and Products*, Technometrics, 4 (1962), pp. 419–420. https://doi.org/10.1080/00401706.1962.10490022.

[89] N. M. Werely, *Analysis and control of linear periodically time varying systems*, PhD thesis, Massachusetts Institute of Technology, 1991.

[90] N. Wiener, *The Homogeneous Chaos*, American Journal of Mathematics, 60 (1938), pp. 897–936. https://doi.org/10.2307/2371268.

[91] T. Woodbury and J. Valasek, *Synthesis and Flight Test of Automatic Landing Controller Using Quantitative Feedback Theory*, Journal of Guidance, Control, and Dynamics, 39 (2016), pp. 1994–2010. https://doi.org/10.2514/1.G001758.

[92] D. Xiu and G. E. Karniadakis, *The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations*, SIAM Journal on Scientific Computing, 24 (2002), pp. 619–644. https://doi.org/10.1137/S1064827501387826.

[93] T. Yamada, K. Ohno, and Y. Ohta, *Comparison between the Hamiltonian Monte Carlo method and the Metropolis-Hastings method for coseismic fault model estimation*, Earth, Planets and Space, 74 (2022). https://doi.org/10.1186/s40623-022-01645-y.

[94] G. Zames, *Feedback and Optimal Sensitivity: Model Reference Transformations, Multiplicative Seminorms, and Approximate Inverses*, IEEE Trans. Automatic Control, 26 (1981), pp. 301–320. https://doi.org/10.1109/TAC.1981.1102603.

[95] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice Hall, 1996.

Appendix A

Combined Standard Deviation of Two Sets

Line 6 of Algorithm 1 computes the total standard deviation at iteration i by combining the standard deviation of the ith noise sample with the total standard deviation at iteration $i-1$. For this task, Equation (6.7) is applied. We derive Equation (6.7) in the following, based on [27]. For $i \in \{1, 2\}$, consider two data sets of size n_i. Denote their individual mean by $\overline{y}_i$ and standard deviation by σ_i. The combined mean of both data sets is

$$\overline{y}_c = \frac{n_1 \overline{y}_1 + n_2 \overline{y}_2}{n_1 + n_2}. \tag{A.1}$$

The combined standard deviation σ_c is further given by

$$\sigma_c^2 = \frac{n_1 \left[\sigma_1^2 + (\overline{y}_1 - \overline{y}_c)^2\right] + n_2 \left[\sigma_2^2 + (\overline{y}_2 - \overline{y}_c)^2\right]}{n_1 + n_2}. \tag{A.2}$$

Equation (6.7) used in line 6 of Algorithm 1 follows by setting the first data set as the first $i-1$ noise samples (seeds) and the second data set as the ith noise sample (seed).